



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

پایان نامه کارشناسی ارشد
گرایش شبکه‌های کامپیوتری

زنجیره‌سازی کارکردهای مجازی سرویس شبکه با در
نظر گرفتن محدودیت منابع مدیریتی

نگارش

پرهام الوانی

استاد راهنما

دکتر بهادر بخشی

شهریور ۱۳۹۸

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع- موجود در پرونده آموزشی- را قرار دهید.

نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

به نام خدا

تاریخ: شهریور ۱۳۹۸

تعهدنامه اصالت اثر

اینجانب پرهام الوانی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

پرهام الوانی

امضا

این رساله هر چند کوچک را تقدیم میکنم به:

. دکتر بخشی که ع سال شاگردی و همکاری با ایشان برای من مایه افتخار بود.

. دوستانم در دانشگاه امیرکبیر که امروز بسیاری شان از پیش مارفته اند اما خاطرشان همواره در یادم می ماند.

. همکارانم در تیم چارلی که نور جدیدی به زندگی من دادند.

سپاس‌گزاری

در اینجا لازم می‌دانم از راهنمایی‌ها و مساعدت‌های استاد عزیز و گرانقدرم جناب آقای دکتر بخشی صمیمانه قدردانی و سپاس‌گزاری نمایم. در ادامه از دوست خوبم بهروز فرکیانی که همواره من را راهنمایی کرده و از پدر و مادرم که همواره من را حمایت کرده‌اند تشکر می‌کنم.

در نهایت جا دارد از دوست، همکار و مدیر خوبم سینا سعیدی، مدیریت فنی تیم منابع مشترک شرکت ایده‌گزین ارتباطات روماک تشکر کنم که بدون حمایت‌های ایشان نگارش این پایان‌نامه ممکن نبود.

پرهم الوانی
شهریور ۱۳۹۸

چکیده

در روش سنتی استقرار سرویس شبکه، ترافیک کاربر باید از تعدادی کارکرد شبکه به ترتیب معینی عبور کند تا یک مسیر پردازش ترافیک ایجاد شود. در حال حاضر این کارکردها به صورت سخت‌افزاری به یکدیگر متصل هستند و ترافیک با استفاده از جداول مسیریابی به سمت آن‌ها هدایت می‌شود. چالش اصلی این روش در این است که استقرار و تغییر ترتیب کارکردها دشوار است. دو فناوری برای پاسخ گویی به این چالش‌ها مطرح شده است: مجازی‌سازی کارکرد شبکه (NFV) و زنجیره‌سازی کارکرد سرویس (SFC). با استفاده از مجازی‌سازی کارکردهای شبکه و اجرای آن‌ها بر روی سرورهای استاندارد با توان بالا، امکان اجرای کارکردها بر روی سخت‌افزارهای عمومی فراهم کرده است تا نیاز به تجهیزات سخت‌افزاری خاص منظوره کاهش یابد. از طرف دیگر SFC امکان تعریف زنجیره کارکردها را ارائه می‌کند که ایجاد و انتخاب مسیرهای متفاوت برای پردازش ترافیک به صورت پویا و بدون ایجاد تغییر در زیرساخت فیزیکی را امکان‌پذیر می‌کند. با توجه به این فناوری‌ها، مسائل تحقیقاتی جدیدی مطرح شده‌اند که از مهم‌ترین آن‌ها می‌توان تخصیص منابع بهینه به سرویس درخواستی کاربر را نام برد. یکی از چالش‌های مهم در زنجیره‌سازی کارکرد سرویس چگونگی جایگذاری کارکردها در شبکه زیرساخت می‌باشد که تا به حال پژوهش‌های زیادی در این حوزه انجام شده است. یکی دیگر از مسائلی که در معماری NFV مطرح است چگونگی مدیریت و نظارت کارکردهای مجازی می‌باشد. تا به حال این دو مساله در کنار یکدیگر مورد مطالعه قرار نگرفته‌اند و این در حالی است که برای ارائه سرویس‌هایی با کیفیت مناسب نیاز است که مدیریت و نظارت بر روی آن‌ها صورت گیرد. در این رساله ما به بررسی همین مساله می‌پردازیم. در اولین گام مساله فوق به صورت خطی صحیح فرمول‌بندی شده و در چهارچوب CPLEX پیاده‌سازی می‌شود. از آنجایی که این مساله NP-Hard می‌باشد نیاز است برای حل آن در زمان مناسب از یک الگوریتم مکاشفه‌ای با پیچیدگی چند جمله‌ای استفاده شود. این رساله الگوریتمی با زمان چند جمله‌ای برای این مساله پیشنهاد می‌دهد و در نهایت آن را با مساله بهینه مقایسه می‌کند. در نتیجه‌ی این مقایسه، الگوریتم پیشنهادی در زمان اجرای کمتر جوابی نزدیک الگوریتم بهینه ارائه می‌کند. الگوریتم پیشنهادی در کنار جایگذاری زنجیره‌ها، نگاشت

منابع مدیریتی را نیز انجام می‌دهد.

واژه‌های کلیدی:

مجازی سازی کارکردهای شبکه، زنجیره سازی کارکردهای مجازی سرویس شبکه، بهینه سازی، بهینه سازی خطی صحیح

فهرست مطالب

صفحه	عنوان
۱	۱ مقدمه
۳	۱-۱ تعریف صورت مساله
۴	۲-۱ اهميت مساله
۴	۳-۱ نوآوری
۵	۴-۱ ساختار گزارش
۶	۲ مفاهيم پايه
۷	۱-۲ مقدمه
۸	۲-۲ مجازی سازی کارکرد شبکه
۱۰	۳-۲ معماری NFV
۱۱	۱-۳-۲ زیرساخت مجازی سازی کارکردهای شبکه یا NFVI
۱۱	۲-۳-۲ کارکردهای مجازی شبکه
۱۲	۳-۳-۲ EM
۱۲	۴-۳-۲ OSS/BSS
۱۲	۵-۳-۲ NFV MANO
۱۴	۴-۲ VNFD
۱۴	۵-۲ سرویس شبکه و اجزای آن
۱۵	۶-۲ موارد کاربرد
۱۵	۷-۲ زنجیره سازی کارکرد سرویس
۱۷	۱-۷-۲ اجزای معماری SFC
۱۸	۸-۲ جمع بندی
۱۹	۳ کارهای مرتبط
۲۰	۱-۳ مقدمه
۲۰	۲-۳ ابعاد مختلف مسائل تحقیقاتی
۲۰	۱-۲-۳ دیدگاه تعریف مساله
۲۱	۳-۳ مرور پژوهش ها
۲۵	۴-۳ جمع بندی

۲۶	۴ تعریف مساله
۲۷	۴-۱ مقدمه
۲۷	۴-۲ مساله
۲۹	۴-۳ مدل سیستم
۲۹	۴-۳-۱ شبکه‌ی زیرساخت
۲۹	۴-۳-۲ منابع مدیریتی
۲۹	۴-۳-۳ انواع
۳۰	۴-۳-۴ زنجیره‌ها
۳۰	۴-۴ فرمول‌بندی
۳۴	۴-۵ مساله‌ی نمونه
۳۷	۴-۶ جمع‌بندی
۳۸	۵ راه‌حل پیشنهادی
۳۹	۵-۱ الگوریتم مکاشفه‌ای
۴۰	۵-۱-۱ Joint Service Deployment - Manager Placement (JSD-MP)
۴۱	۵-۱-۲ enhanced JSD-MP (eJSD-MP)
۴۳	۵-۲ جمع‌بندی
۴۴	۶ ارزیابی
۴۵	۶-۱ مقدمه
۴۵	۶-۲ محیط ارزیابی
۴۶	۶-۳ معیارهای ارزیابی
۴۷	۶-۳-۱ نسبت سود به هزینه
۴۷	۶-۳-۲ سود
۴۷	۶-۳-۳ تعداد زنجیره‌های پذیرفته شده
۴۸	۶-۳-۴ تعداد VNFM‌های استفاده شده
۴۸	۶-۴ محیط ارزیابی
۵۰	۶-۵ نتایج ارزیابی
۵۰	۶-۵-۱ زمان حل بهینه
۵۱	۶-۵-۲ نسبت سود به هزینه
۵۱	۶-۵-۳ زنجیره‌ها در توپولوژی FatTree
۵۳	۶-۵-۴ زنجیره‌ها در توپولوژی USnet
۵۶	۶-۶ ارزیابی زمان اجرا

۵۷	۷-۶ جمع‌بندی
۵۸	۷ نتیجه‌گیری و کارهای آینده
۵۹	۱-۷ جمع‌بندی و نتیجه‌گیری
۶۰	۱-۱-۷ کارهای آینده
۶۱	منابع و مراجع
۶۳	واژه‌نامه

شکل	فهرست اشکال	صفحه
۱-۲	رویکرد NFV	۸
۲-۲	معماری مجازی سازی کارکردهای شبکه	۱۰
۳-۲	یک سرویس شبکه شامل یک گراف جلورانی	۱۴
۴-۲	گراف VNF-FG متناظر با شکل ۳-۲	۱۵
۵-۲	مجازی سازی CPE	۱۶
۶-۲	مجازی سازی زیرساخت LTE	۱۶
۷-۲	معماری SFC	۱۷
۱-۴	زنجیره های مساله ی نمونه	۳۴
۲-۴	توپولوژی زیرساخت مساله ی نمونه	۳۴
۱-۵	مدل سازی با گراف چندگامی [۴]	۳۹
۱-۶	توپولوژی ساختاریافته FatTree	۴۶
۲-۶	توپولوژی تصادفی USnet	۴۶
۳-۶	شکاف بهینه الگوریتم بهینه بر اساس زمان اجرا (بر حسب دقیقه)	۵۰
۴-۶	کارآیی الگوریتم های eJSD-MP و JSD-MP در نسبت های مختلف سود	
۵۱	به هزینه	
۵-۶	سود نهایی الگوریتم های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree	۵۲
۶-۶	تعداد زنجیره های پذیرفته شده الگوریتم های بهینه، eJSD-MP و JSD-MP	
۷-۶	تعداد VNFM های الگوریتم های بهینه، eJSD-MP و JSD-MP برای توپولوژی	
۵۴	FatTree	
۸-۶	سود نهایی الگوریتم های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet	۵۴
۹-۶	تعداد زنجیره های پذیرفته شده الگوریتم های بهینه، eJSD-MP و JSD-MP	
۵۵	MP برای توپولوژی USnet	
۱۰-۶	تعداد VNFM های الگوریتم های بهینه، eJSD-MP و JSD-MP برای توپولوژی	
۵۶	USnet	

صفحه	فهرست جداول	جدول
۲۴	مقایسه مقالات پذیرش زنجیره‌های کارکرد سرویس	۱-۳
۳۰	پارامترهای مساله	۱-۴
۳۱	متغیرهای تصمیم‌گیری مساله (قسمت اول)	۲-۴
۳۳	متغیرهای تصمیم‌گیری مساله (قسمت دوم)	۳-۴
۳۵	نیازمندی نمونه‌های مساله‌ی نمونه	۴-۴
۳۵	مشخصات سرورهای زیرساخت مساله‌ی نمونه	۵-۴
۳۵	نتایج مساله‌ی نمونه	۶-۴
۵۳	سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree	۱-۶
۵۵	سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet	۲-۶

فهرست نمادها

نماد	مفهوم
$V_{i,F}^{SFC}$	مجموعه گره‌های گراف زنجیره‌ی i ام
E_i^{SFC}	مجموعه یال‌های گراف زنجیره‌ی i ام
V_S^{PN}	مجموعه گره‌های گراف زیرساخت
E_S^{PN}	مجموعه یال‌های گراف زیرساخت
x_h	متغیر باینری که نشان می‌دهد زنجیره‌ی h ام پذیرفته شده است یا خیر
y_{wk}	تعداد نمونه‌هایی از نوع k که روی سرور فیزیکی w فعال شده‌اند
z_{vw}^k	متغیر باینری که نشان می‌دهد نمونه‌ی v از نوع k روی سرور فیزیکی w جایگذاری شده است یا خیر
\bar{y}_w	تعداد نمونه‌هایی از VNFM که روی سرور w فعال شده‌اند
\bar{z}_{hw}	متغیر باینری که نشان می‌دهد زنجیره‌ی h توسط VNFM‌ای که روی سرور w قرار گرفته است مدیریت می‌شود یا خیر
$\tau_{ij}^{(u,v)}$	متغیر باینری که نشان می‌دهد یال مجازی بین نمونه‌های u و v برای نگاشت از یال فیزیکی بین گره‌های i و j استفاده می‌کند یا خیر
$\bar{\tau}_{ij}^v$	متغیر باینری که نشان می‌دهد برای نگاشت ارتباط مدیریتی نمونه‌ی v از یال فیزیکی بین گره‌های i و j استفاده شده است یا خیر

فصل اول

مقدمه

راه‌اندازی و استقرار سرویس در صنعت مخابرات به طور سنتی بر این اساس است که اپراتورهای شبکه سخت‌افزارهای اختصاصی فیزیکی و تجهیزات لازم برای هر کارکرد در سرویس را در زیرساخت خود مستقر کنند. فراهم کردن نیازمندی‌هایی مانند پایداری و کیفیت بالا منجر به اتکای فراهم کنندگان سرویس بر سخت‌افزارهای اختصاصی می‌شود. این در حالی است که نیازمندی کاربران به سرویس‌های متنوع و عموماً با عمر کوتاه و نرخ بالای ترافیک افزایش یافته است. بنابراین فراهم کنندگان سرویس‌ها باید مرتباً و به صورت پیوسته تجهیزات فیزیکی جدید را خریده، انبارداری کرده و مستقر کنند. تمام این عملیات باعث افزایش هزینه‌های فراهم کنندگان سرویس می‌شود. با افزایش تجهیزات، پیدا کردن فضای فیزیکی برای استقرار تجهیزات جدید به مرور دشوارتر می‌شود. علاوه بر این باید افزایش هزینه و تاخیر ناشی از آموزش کارکنان برای کار با تجهیزات جدید را نیز در نظر گرفت. بدتر این که هر چه نوآوری سرویس‌ها و فناوری شتاب بیشتری می‌گیرد، چرخه عمر سخت‌افزارها کوتاه‌تر می‌شود که مانع از ایجاد نوآوری در سرویس‌های شبکه می‌شود.

در روش سنتی استقرار سرویس شبکه، ترافیک کاربر باید از تعدادی کارکرد شبکه به ترتیب معینی عبور کند تا یک مسیر پردازش ترافیک ایجاد شود. در حال حاضر این کارکردها به صورت سخت‌افزاری به یکدیگر متصل هستند و ترافیک با استفاده از جداول مسیریابی به سمت آن‌ها هدایت می‌شود. چالش اصلی این روش در این است که استقرار و تغییر ترتیب کارکردها دشوار است. به عنوان مثال، به مرور زمان با تغییر شرایط شبکه نیازمند تغییر همبندی و یا مکان کارکردها برای سرویس‌دهی بهتر به کاربران هستیم که نیاز به جابه‌جایی کارکردها و تغییر جداول مسیریابی دارد. در روش سنتی، این کار، سخت و هزینه‌بر است که ممکن است خطاهای بسیاری در آن رخ دهد. از جنبه دیگر، تغییر سریع سرویس‌های مورد نظر کاربران نیازمند تغییر سریع در ترتیب کارکردها است که در روش فعلی این تغییرات به سختی صورت می‌گیرد. بنابراین اپراتورهای شبکه نیاز به شبکه‌های قابل برنامه‌ریزی و ایجاد زنجیره سرویس کارکردها به صورت پویا پیدا کرده‌اند. در سال‌های اخیر دو تکنولوژی شبکه‌های نرم‌افزارمحور و مجازی‌سازی شبکه بسیار مورد توجه قرار گرفته‌اند. پیش‌تر در ارائه سرویس‌های شبکه، از سخت‌افزارهای اختصاصی که توسط سازندگان اختصاصی ارائه می‌شد و به آن‌ها middle box گفته می‌شد استفاده می‌گشت. تنوع و تعداد رو به افزایش سرویس‌های جدیدی که توسط کاربران تقاضا می‌گردد باعث هزینه‌های زیاد برای خرید و نگهداری middle boxها توسط اپراتورها شده است. به تازگی فراهم‌آوردندگان شبکه شروع به حرکت به سوی مجازی‌سازی و نرم‌افزاری کردن بسترهای شبکه کرده‌اند. به این ترتیب آن‌ها قادر خواهند بود سرویس‌های نوآورانه‌ای به کاربران ارائه دهند. این روند به سرویس‌دهندگان اجازه می‌دهد که ارائه سرویس‌های

دلخواه‌شان وابسته به سخت‌افزارهای اختصاصی نباشد و هزینه‌های راه‌اندازی و نگهداری فراهم آوردندگان سرویس را کاهش می‌دهد. با نرم‌افزاری‌سازی کارکردها، وابستگی آن‌ها به سخت‌افزار اختصاصی کاهش یافته و به سرعت می‌توان آن‌ها را افزایش/کاهش مقیاس داد. مجازی‌سازی کارکردهای شبکه و زنجیره‌سازی کارکرد سرویس راهکاری‌هایی هستند که برای همین منظور پیشنهاد شده‌اند.

ایده‌ی اصلی مجازی‌سازی توابع شبکه جداسازی تجهیزات فیزیکی شبکه از کارکردهایی می‌باشد که بر روی آن‌ها اجرا می‌شوند. به این معنی که یک کارکرد شبکه مانند دیوار آتش می‌تواند بر روی سرورهای HVS^۱ به عنوان یک نرم‌افزار ساده مستقر شود. با این روش یک سرویس می‌تواند با استفاده از کارکردهای مجازی شبکه‌ای که می‌توانند به صورت نرم‌افزاری پیاده‌سازی شده و روی یک یا تعدادی میزبان استاندارد فیزیکی اجرا شوند، استقرار یابد. کارکردهای مجازی شبکه‌ای می‌توانند در مکان‌های مختلف بازمان‌یابی یا نمونه‌سازی شوند بدون آنکه نیاز به خریداری و نصب تجهیز جدیدی باشد. [۱۲]

۱-۱ تعریف صورت مساله

مساله‌ی جاسازی کارکردهای مجازی شبکه یکی از چالش‌های مهم در تخصیص منابع به زنجیره‌های کارکرد سرویس می‌باشد. مساله جاسازی کارکردهای مجازی شبکه به دو زیر مساله‌ی نگاشت گره‌های مجازی و نگاشت یال‌های تقسیم می‌شود، که می‌بایست به صورت توامان در نظر گرفته شوند.

البته محدودیت‌های زیادی وجود دارد که باید هنگام نگاشت در نظر گرفته شود. منابع فیزیکی انتخاب شده از شبکه زیرساخت باید نیازمندی‌های کارکرد شبکه مجازی را تامین کنند. به عنوان مثال قدرت پردازشی کارکردهای مجازی باید کمتر یا مساوی با قدرت پردازشی گره فیزیکی باشد که نگاشت روی آن انجام شده است. یا نیازمندی یک کارکرد به یک سرور فیزیکی خاص برای اجرا که می‌بایست مدنظر قرار داده شود.

علاوه بر این، مجموعه‌ای از محدودیت‌ها وجود دارد که مختص زنجیره‌های کارکرد سرویس می‌باشد. یکی از این موارد وجود VNFM در این شبکه‌ها می‌باشد که به علت اهمیت میزان تاخیر ارتباط بین کارکرد مجازی شبکه و VNFM می‌بایست در مکان مناسبی جایابی شود. بنابراین زیرمساله‌ی جدیدی به مساله‌ی اصلی اضافه می‌شود.

^۱ High Volume Server

۲-۱ اهمیت مساله

مساله‌ی جاسازی زنجیره‌های کارکرد سرویس از اهمیت زیادی برخوردار است و پژوهش‌های زیادی بر روی آن صورت پذیرفته است. در کنار این، مساله مدیریت و نظارت این زنجیره‌ها نیز مطرح است که این پروژه برای اولین بار این موضوع را نیز مدنظر قرار داده است که باعث می‌شود اهمیت مساله دو چندان شود.

امروزه هزینه‌ی زیادی صرف نظارت مراکز داده می‌شود و نظارت برای سرویس‌ها امر بسیار مهمی است. در بسیاری از موارد نظارت امری است که در آینده به آن فکر می‌شود و آسیب‌های زیادی در پی دارد. این پژوهش قصد دارد نیازمندی نظارت برای سرویس‌ها را در زمان نگاشت آن‌ها مدنظر قرار دهد تا از خسارت‌های نبود نظارت برای سرویس‌ها جلوگیری کند.

۳-۱ نوآوری

ایده‌ی اصلی این پژوهش، ارائه‌ی یک راه‌حل جامع و کامل که تمامی ابعاد مساله‌ی جاسازی زنجیره‌های کارکرد سرویس را در برگیرد، است. در واقع در این مساله علاوه بر در نظر گرفتن ابعاد اصلی مساله‌ی جاسازی مکانیزم کنترل پذیرش، قابل اعمال بودن راه‌حل به توپولوژی‌های مختلف و وجود محدودیت‌های گره و یال، ابعاد دیگری نیز در نظر گرفته شده است. به علت وجود VNFM به عنوان یک گره خاص و اهمیت تاخیر اتصالات کارکرد مجازی شبکه و VNFM، یک مرحله جایابی و نگاشت به مساله‌ی اصلی اضافه شده است. در ادامه محدودیت‌هایی برای اتصالات بین کارکردهای مجازی شبکه و VNFM‌ها در نظر گرفته شده است و فرض شده است برای مدیریت تعداد مشخصی از کارکردهای مجازی نیاز به تهیه مجوز با هزینه‌ای مشخص است.

- یکی از نوآوری‌های اصلی این رساله تعریف مساله‌ای با در نظر گرفتن نیازمندی‌های مدیریتی است که به مدیر سیستم امکان می‌دهد تا تمامی سیاست‌های مورد نظرش را در مساله اعمال و تنظیم نماید. مساله‌ی جایگذاری زنجیره‌های کارکرد سرویس مساله‌ای بسیار مهم در این حوزه است که این رساله این مساله را به صورت توانمند با محدودیت‌های منابع مدیریتی بررسی کرده و آن را در قالب یک مساله‌ی کنترل پذیرش به صورت خطی صحیح فرمول‌بندی می‌کند.

- نوآوری دیگری این رساله ارائه‌ی یک روش مکاشفه‌ای بر پایه ایده‌ی الگوریتم [۴] می‌باشد که در ادامه زمان اجرا و سود نهایی آن را نیز بهبود می‌دهد.

۴-۱ ساختار گزارش

در ادامه معماری NFV را معرفی می‌کنیم و به چالش‌هایی که در MANO وجود دارد می‌پردازیم. در فصل سوم کارهای مرتبط مرور می‌شوند و در فصل چهارم مساله تعریف شده به همراه فرمول‌بندی بیان می‌گردد. در فصل پنجم در مورد راه‌حل پیشنهادی برای مساله بحث خواهد شد. در آخر در فصل ششم راه‌حل پیشنهادی ارزیابی می‌گردد و نتایج در فصل هفتم جمع‌بندی می‌گردند.

فصل دوم

مفاهیم پایه

۱-۲ مقدمه

راه‌اندازی و استقرار سرویس در صنعت مخابرات به طور سنتی بر این اساس است که اپراتورهای شبکه سخت‌افزارهای اختصاصی فیزیکی و تجهیزات لازم برای هر کارکرد در سرویس را در زیرساخت خود مستقر کنند. فراهم کردن نیازمندی‌هایی مانند پایداری و کیفیت بالا منجر به اتکای فراهم‌کنندگان سرویس بر سخت‌افزارهای اختصاصی می‌شود. این در حالی است که نیازمندی کاربران به سرویس‌های متنوع و عموماً با عمر کوتاه و نرخ بالای ترافیک افزایش یافته است. بنابراین فراهم‌کنندگان سرویس‌ها باید مرتباً و به صورت پیوسته تجهیزات فیزیکی جدید را خریده، انبارداری کرده و مستقر کنند. تمام این عملیات باعث افزایش هزینه‌های فراهم‌کنندگان سرویس می‌شود. با افزایش تجهیزات، پیدا کردن فضای فیزیکی برای استقرار تجهیزات جدید به مرور دشوارتر می‌شود. علاوه بر این باید افزایش هزینه و تاخیر ناشی از آموزش کارکنان برای کار با تجهیزات جدید را نیز در نظر گرفت. بدتر این که هر چه نوآوری سرویس‌ها و فناوری شتاب بیشتری می‌گیرد، چرخه عمر سخت‌افزارها کوتاه‌تر می‌شود که مانع از ایجاد نوآوری در سرویس‌های شبکه می‌شود.

در روش سنتی استقرار سرویس شبکه، ترافیک کاربر باید از تعدادی کارکرد شبکه به ترتیب معینی عبور کند تا یک مسیر پردازش ترافیک ایجاد شود. در حال حاضر این کارکردها به صورت سخت‌افزاری به یکدیگر متصل هستند و ترافیک با استفاده از جداول مسیریابی به سمت آن‌ها هدایت می‌شود. چالش اصلی این روش در این است که استقرار و تغییر ترتیب کارکردها دشوار است. به عنوان مثال، به مرور زمان با تغییر شرایط شبکه نیازمند تغییر همبندی و یا مکان کارکردها برای سرویس‌دهی بهتر به کاربران هستیم که نیاز به جابه‌جایی کارکردها و تغییر جداول مسیریابی دارد. در روش سنتی این کار سخت و هزینه‌بر است که ممکن است خطاهای بسیاری در آن رخ دهد. از جنبه‌ی دیگر، تغییر سریع سرویس‌های مورد نظر کاربران نیازمند تغییر سریع در ترتیب کارکردها است که در روش فعلی این تغییرات به سختی صورت گیرد. بنابراین اپراتورهای شبکه نیاز به شبکه‌های قابل برنامه‌ریزی و ایجاد زنجیره سرویس کارکردها به صورت پویا پیدا کرده‌اند. دو فناوری برای پاسخ‌گویی به این چالش‌ها مطرح شده است:

- مجازی‌سازی کارکرد شبکه یا NFV

- زنجیره‌سازی کارکردهای سرویس یا SFC

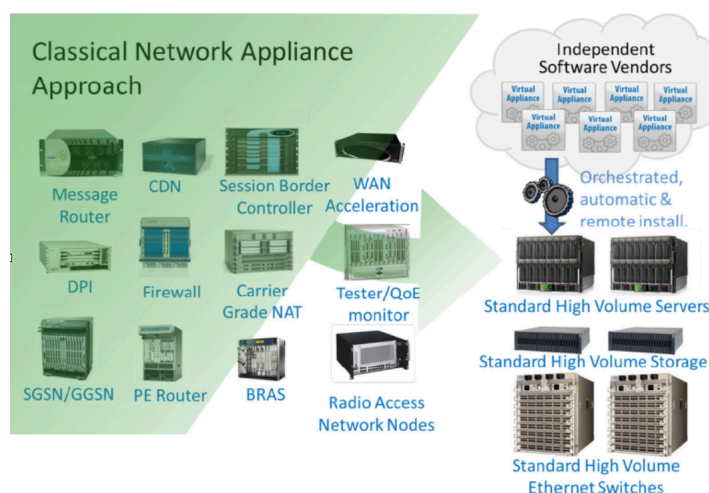
با استفاده از مجازی‌سازی کارکردهای شبکه و اجرای آن‌ها بر روی سرورهای استاندارد با توان بالا، امکان اجرای کارکردها بر روی سخت‌افزارهای عمومی فراهم شده است تا نیاز به تجهیزات سخت‌افزاری خاص منظوره کاهش یابد. از طرف دیگر SFC امکان تعریف زنجیره

کارکردها را ارائه می‌کند که ایجاد و انتخاب مسیرهای متفاوت برای پردازش ترافیک به صورت پویا و بدون ایجاد تغییر در زیرساخت فیزیکی را امکان‌پذیر می‌کند با توجه به این فناوری‌ها، مسائل تحقیقاتی جدیدی مطرح شده‌اند که از مهم‌ترین آن‌ها می‌توان تخصیص منابع بهینه به سرویس درخواستی کاربر را نام برد.

از آنجایی که از مفاهیم این فناوری‌ها برای طراحی و تعریف مساله در این رساله استفاده شده است، نیازمند آشنایی با مفاهیم ابتدایی و اصول اولیه آن‌ها خواهیم بود. بنابراین در این فصل به صورت خلاصه اجزای این فناوری‌ها را مرور خواهیم کرد و کاربردها، چالش‌ها و مسائل تحقیقاتی که در هر یک از این معماری‌ها وجود دارد را مورد بررسی قرار خواهیم داد.

۲-۲ مجازی‌سازی کارکرد شبکه

مجازی‌سازی کارکرد شبکه اصل جداسازی کارکرد شبکه به وسیله انتزاع سخت‌افزاری مجازی از سخت‌افزاری است که بر روی آن اجرا می‌شود. هدف مجازی‌سازی کارکرد شبکه تغییر روش اپراتورهای شبکه در طراحی شبکه با تکامل مجازی‌سازی استاندارد فناوری اطلاعات به منظور جمع‌تجهیزات شبکه در سرورهای استاندارد، سویچ‌ها و ذخیره‌سازها با توان بالا است. یک سرور استاندارد با توان بالا سروری است که توسط اجزای استاندارد شده IT، مانند معماری x86، ساخته شده و در تعداد بالایی، مانند میلیون، فروخته می‌شود. ویژگی اصلی این سرورها این است که اجزای آن‌ها به راحتی از فروشندگان مختلف قابل خریداری و تعویض است. این تجهیزات می‌توانند در مراکز داده، گره‌های شبکه، یا مکان کاربران انتهایی قرار بگیرند. این روند در شکل ۱-۲ نیز توصیف شده است.



شکل ۱-۲: رویکرد NFV

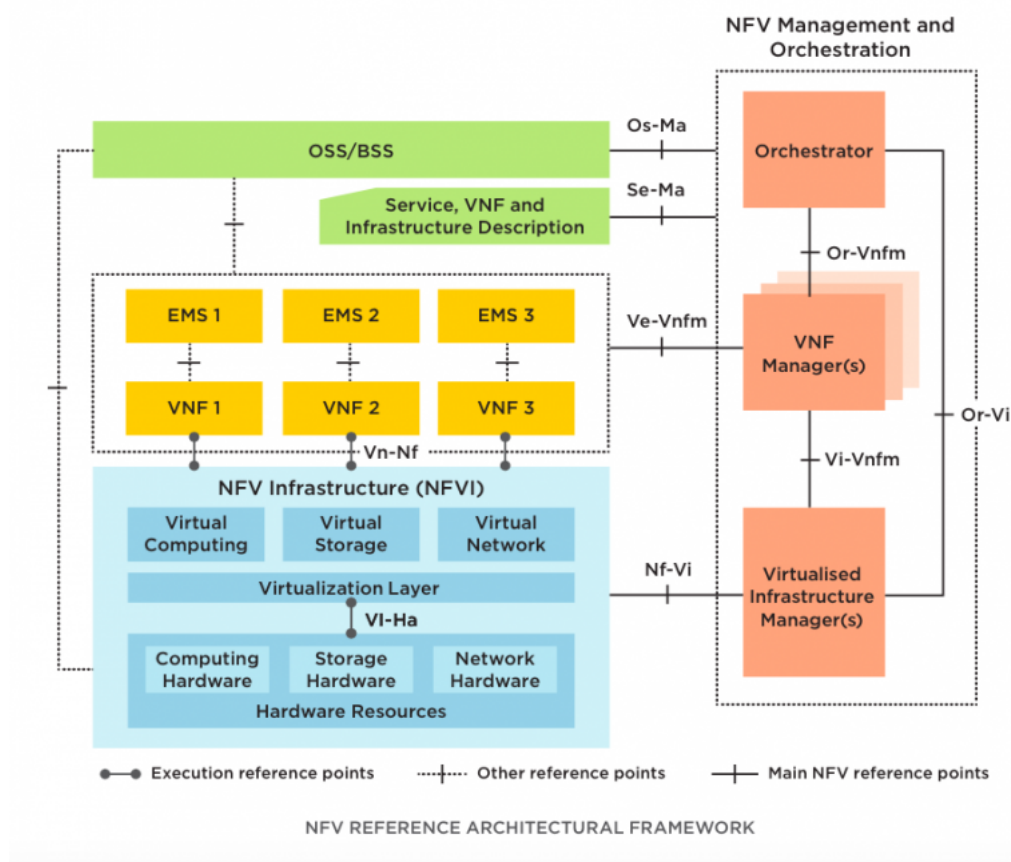
با استفاده از NFV، انواع کارکردهای شبکه مانند دیواره آتش و NAT را می‌توان به صورت یک برنامه نرم‌افزاری از فروشندگان مختلف تهیه کرد و آن‌ها را بر روی سرورهای با توان بالا اجرا کرد که نیاز به نصب تجهیزات خاص منظوره و جدید را برطرف می‌سازد. مزایا و اهداف اساسی که NFV برای تحقق و دستیابی به آنها شکل گرفته است عبارتند از:

- کاهش هزینه‌های تجهیزات و مصرف انرژی از طریق تجمیع کارکردها بر روی سرورها و در نتیجه کاهش تعداد تجهیزات
- کاهش نیاز به آموزش کارکنان، افزایش دسترسی پذیری به سخت افزار و کاهش زمان بازیابی از خرابی سخت افزار به علت استفاده از سخت افزارهای استاندارد و عمومی
- افزایش سرعت عرضه محصول به بازار با کوتاه کردن چرخه نوآوری و تولید. در واقع NFV به اپراتورهای شبکه کمک میکند تا چرخه بلوغ محصول را به اندازه قابل توجهی کاهش دهند.
- امکان پذیر بودن تعریف سرویس مورد نظر بر اساس نوع مشتری یا محل جغرافیایی. مقیاس سرویس‌ها میتواند به سرعت، بر اساس نیاز، گسترش یا کاهش یابد.
- تشویق به ایجاد نوآوری و ارائه سرویس‌های جدید و دریافت جریان‌های درآمدی تازه با سرعت بالا و ریسک پایین.
- افزایش توانایی مقابله با خرابی کارکردها، قابلیت به اشتراک گذاری منابع بین کارکردها و پشتیبانی از چند مشتری

سازمان‌های استانداردگذاری متعددی در استانداردسازی فناوری NFV دخیل هستند که شاخص‌ترین آن‌ها موسسه استانداردهای مخابراتی اروپا (ETSI) است. در اواخر سال ۲۰۱۲، ETSI NFV ISG توسط هفت اپراتور جهانی شبکه به منظور ارتقا ایده مجازی‌سازی کارکرد شبکه تأسیس شد. NFV ISG تبدیل به یک بستر صنعتی اصلی برای توسعه چارچوب معماری NFV و نیازمندی‌های آن شده است و اکنون بیش از ۲۵۰ سازمان با آن همکاری می‌کنند. اسناد معماری NFV به صورت عمومی و رایگان توسط ETSI NFV ISG منتشر می‌شود. ما در این رساله برای توصیف معماری NFV از اسناد ارائه شده این سازمان استفاده می‌کنیم.

۳-۲ معماری NFV

در این بخش مؤلفه‌های تشکیل‌دهنده معماری NFV شرح داده می‌شوند. هر یک از اجزای معماری می‌توانند توسط تولیدکنندگان متفاوتی تأمین شوند و به وسیله واسطه‌هایی که توسط معماری NFV توصیف شده‌اند با یکدیگر در ارتباط باشند. بنابراین معماری NFV توصیف شده توسط ETSI راه‌حلی با قابلیت مشارکت و هماهنگی چندین تولیدکننده مختلف را دارد. با توجه به استاندارد ETSI معماری NFV از سه عنصر کلیدی تشکیل شده است. زیرساخت مجازی‌سازی کارکردهای شبکه، کارکردهای مجازی شبکه‌ای و NFV MANO. این اجزا در شکل ۲-۲ نمایش داده شده‌اند.



شکل ۲-۲: معماری مجازی‌سازی کارکردهای شبکه

- **NFVI**: شامل منابع سخت افزاری و نرم‌افزاری لازم برای اجرای VNFها
- **Service**: شامل VNFها که کارکردهای شبکه را پیاده‌سازی کرده‌اند، EMS برای مدیریت VNFها و OSS/BSS برای ارتباط با سیستم‌های مدیریت سنتی
- **MANO**: که وظیفه مدیریت و هماهنگی سرویس‌ها و تخصیص منابع را برعهده دارد و از سه بخش NFVO، VIM و VNFM تشکیل شده است.

۲-۳-۱ زیرساخت مجازی سازی کارکردهای شبکه یا NFVI

زیرساخت مجازی سازی کارکردهای شبکه ترکیبی از منابع نرم افزاری و سخت افزاری است که محیطی برای نصب کارکردهای مجازی شبکه فراهم می آورد. منابع سخت افزاری شامل منابع محاسباتی، ذخیره سازها و شبکه (شامل لینک ها و گره ها) هستند که پردازش، ذخیره سازی و ارتباط را برای کارکردهای مجازی شبکه فراهم می آورند. منابع مجازی انتزاعی از منابع شبکه ای، پردازشی و ذخیره سازی هستند. به وسیله انتزاع از طریق لایه ی مجازی سازی (بر پایه ی hypervisor) منابع سخت افزاری در اختیار کارکردهای مجازی قرار می گیرند که این منابع شامل منابع محاسباتی، شبکه ای و ذخیره سازی می باشند.

در مراکز داده ای ممکن است منابع پردازشی و ذخیره سازی تحت عنوان یک یا چند ماشین مجازی نمایش داده شوند در حالی که شبکه های مجازی از لینک ها و گره های مجازی تشکیل می شوند. شبکه های مجازی پیش از بحث مجازی سازی کارکردهای شبکه مدنظر بوده اند و روی آن ها کار شده است. در واقع از شبکه های مجازی در مراکز داده ای جهت فراهم آوردن شبکه های مختلف و مجزا که به کاربران مختلفی تعلق دارند استفاده شده است. راه حل های مختلفی برای پیاده سازی این شبکه ها وجود دارد. در بحث مجازی سازی کارکردهای شبکه، زیرساخت ارتباطی مورد نیاز برای کارکردهای مجازی از طریق همین شبکه های مجازی فراهم آورده می شود. یعنی مسائلی که پیش تر در بحث جایگذاری شبکه های مجازی مطرح بود امروز جزئی از مسائل جایگذاری زنجیره های کارکرد سرویس می باشند.

۲-۳-۲ کارکردهای مجازی شبکه

یک کارکرد شبکه، یک بلوک عملیاتی در زیرساخت شبکه است که عملکرد رفتاری و رابط های ارتباط با خارج خوش تعریف دارد. مثال هایی از کارکردهای شبکه می تواند شامل DHCP یا firewall و ... باشد. با این توضیحات، کارکرد مجازی شبکه، پیاده سازی یک کارکرد شبکه است که می تواند روی منابع مجازی شده اجرا شود. از هر کارکرد شبکه می توان نمونه سازی کرده و چند نمونه را در شبکه مستقر ساخت. این نمونه ها می توانند برای سرویس دهی به زنجیره های مختلف استفاده شوند. از آنجایی که هر نمونه توان پردازشی محدودی دارد با افزایش تعداد نمونه ها می توان توان پردازشی یک کارکرد را نیز افزایش داد.

EM ۳-۳-۲

این مولفه کارکردهای FCAPS^۱ را برای VNF ها انجام می‌دهد که شامل مدیریت خطا، پیکربندی، امنیت، حسابداری و کارایی برای کارکردی است که VNF ارائه می‌دهد. این مولفه ممکن است آگاه از مجازی بودن کارکرد، باشد و با همکاری VNFM عملکردهای خودش را انجام بدهد.

OSS/BSS ۴-۳-۲

این مولفه، ترکیبی از سایر بخش‌های عملکردهای اپراتور است که در چارچوب معماری NFV ارائه شده از طرف ETSI قرار نمی‌گیرند. به عنوان مثال می‌تواند شامل مدیریت سیستم‌های Legacy باشد.

NFV MANO ۵-۳-۲

بر اساس چهارچوب پیشنهادی ETSI وظیفه‌ی NFV MANO فراهم آوردن کارکردهای لازم برای تدارک فرآیندهای مشابه مانند تنظیم کردن و ... کارکردهای مجازی شبکه است. NFV MANO شامل هماهنگ‌کننده و مدیریت‌کننده چرخه‌ی زندگی منابع سخت‌افزاری و نرم‌افزاری که مجازی‌سازی زیرساخت را پشتیبانی می‌کنند، است. هر زنجیره نیاز دارد که حداقل توسط یک VNFM مدیریت شود تا مثلاً خطاهای آن را تحت نظر قرار دهد و در صورت نیاز در قسمت دیگری از شبکه استقرار یابد. مساله‌ی جایگذاری زنجیره‌ها بسیار مورد مطالعه قرار گرفته است، اما در این بین توجه لازم به نیاز این زنجیره‌ها به یک VNFM صورت پذیرفته است.

VNFO بخشی از مولفه MANO است که وظیفه تخصیص منابع به سرویس را برعهده دارد. یکی از مهم‌ترین اجزای سرویس گراف VNF-FG است که بیانگر VNF های سرویس و ارتباطات بین آن‌ها است. وظیفه اصلی مولفه NFVO ایجاد نمونه از سرویس و مدیریت چرخه حیات آن است. ایجاد نمونه از سرویس شامل ایجاد نمونه از VNF های تشکیل‌دهنده آن و ایجاد ارتباط بین نمونه‌ها است. سایر وظایف مولفه VNFO به شرح زیر است:

- مدیریت چرخه حیات سرویس شبکه
- مدیریت و هماهنگی منابع مورد نیاز NFVI بین چندین VIM
- مدیریت منابع و ایجاد نمونه از VNF ها با هماهنگی VNFM
- مدیریت منابع و نمونه‌سازی VNFM

^۱Fault, Config, Accounting, Performance, Security

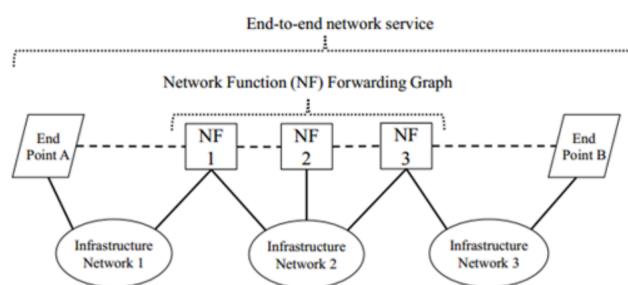
- مدیریت همبندی نمونه ساخته شده از سرویس شبکه مانند ایجاد، حذف و به روز رسانی VNF-FG
- مدیریت قالب‌های استقرار سرویس شبکه و VNF‌ها مانند اعتبارسنجی قالب‌ها
- همچنین این مولفه مسئولیت مشخص کردن مکان فیزیکی نمونه‌های ایجاد شده از VNF‌ها را برعهده دارد.
- مولفه‌ی VNFM مسئولیت مدیریت چرخه حیات نمونه‌های ایجاد شده از VNF‌ها را برعهده دارد. بنابراین فرض می‌شود هر نمونه ایجاد شده از هر VNF، به یک VNFM اختصاص یافته است. مهم‌ترین وظایف این مولفه به شرح زیر است:
 - پیکربندی و نمونه‌سازی از VNF‌ها
 - گسترش و یا کاهش مقیاس‌پذیری افقی یا عمودی برای نمونه‌های ایجاد شده از VNF‌ها
 - مدیریت نمونه‌های ایجاد شده شامل تغییرات، به روزرسانی برنامه‌ها و خاتمه دادن به نمونه‌ها
 - مولفه VNFM با استفاده از VNFD، از VNF نمونه ایجاد می‌کند و مدیریت چرخه حیات آن را انجام می‌دهد. منابع پردازشی، محاسباتی و شبکه مطابق با توصیفات گفته شده در VNFD به نمونه‌های آن اختصاص می‌یابند.
 - مولفه VIM مسئولیت کنترل و مدیریت منابع محاسباتی، ذخیره‌سازی و شبکه‌ای، معمولاً در حوزه یک اپراتور، را برعهده دارد. مهم‌ترین وظایف این مولفه عبارتند از:
 - هماهنگی تخصیص، ارتقا و آزادسازی منابع NFVI شامل بهینه‌سازی استفاده از منابع و مدیریت انجمنی منابع مجازی و فیزیکی. بنابراین VIM اطلاعات Inventory تخصیص منابع مجازی به منابع فیزیکی را نگهداری می‌کند.
 - پشتیبانی از مدیریت VNF-FG به وسیله ایجاد و نگهداری لینک‌های مجازی، شبکه‌های مجازی زیرشبکه‌ها و پورت‌ها
 - مدیریت اطلاعات Inventory سخت افزارها و نرم افزارها و کشف قابلیت‌ها و ویژگی‌های آنها
 - مدیریت ظرفیت منابع مجازی مانند نسبت منابع مجازی به حقیقی
 - مدیریت تصویرهای نرم‌افزاری، مانند تصاویر VNF‌ها، که ممکن است توسط سایر مولفه‌های MANO هم مورد استفاده قرار بگیرند.
 - جمع‌آوری اطلاعات کارایی و خطا از منابع سخت‌افزاری و نرم‌افزاری

۴-۲ VNFD

هر VNF توسط توصیف‌گر مربوط به آن که نیازمندی‌های استقرار و رفتاری آن را مشخص می‌کند توصیف می‌شود. مولفه VNFM از VNFD در فرایند نمونه‌سازی VNF ها و مدیریت چرخه حیات آن‌ها استفاده می‌کند. همچنین این اطلاعات توسط مولفه VNFO برای ایجاد مدیریت و هماهنگی سرویس شبکه نیز استفاده می‌شود. VNFD شامل شاخص‌های کارایی است که می‌تواند توسط VNFM نیز مورد استفاده قرار بگیرد. در VNFD ارتباطات داخلی و واسط‌ها نیز توصیف می‌شوند که برای ایجاد لینک‌های مجازی بین مولفه‌های VNFC و یا ارتباط بین VNF با سایر VNF ها مورد استفاده قرار می‌گیرد. VNFD همچنین شامل قالب‌های استقرار VNF به همراه نیازمندی منابع برای هر قالب است.

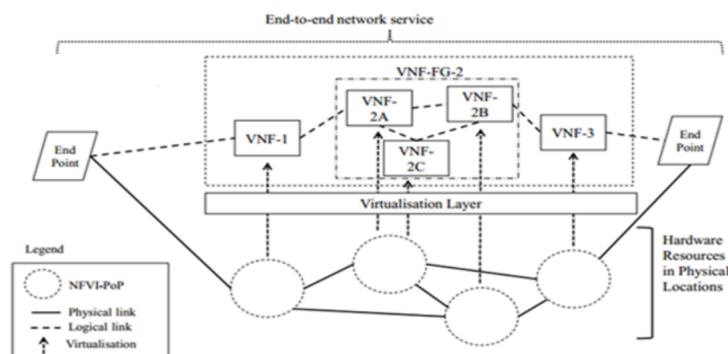
۵-۲ سرویس شبکه و اجزای آن

یک سرویس شبکه را می‌توان به صورت یک گراف جلورانی از کارکردهای شبکه (NF-FG) که به یکدیگر از طریق زیرساخت شبکه متصل هستند دید. کارکردهای شبکه می‌تواند توسط یک یا چند اپراتور ارائه شده باشند. نقاط انتهایی سرویس را می‌توان به صورت گره‌های گراف و ارتباطات میان کارکردها را توسط لینک‌های گراف مدل سازی کرد که لینک‌های گراف می‌توانند، یک طرفه یا دو طرفه، چند پخشی یا همه پخشی باشند. مثالی از یک سرویس شبکه در شکل ۳-۲ نمایش داده شده است. در این شکل، یک سرویس شبکه انتها به انتها از طریق نقاط انتهایی A و B ایجاد شده که شامل یک NF-FG داخلی است. این NF-FG خود شامل سه کارکرد شبکه است که به یکدیگر متصل هستند.



شکل ۳-۲: یک سرویس شبکه شامل یک گراف جلورانی

در صورتی که در یک NF-FG حداقل یکی از این کارکردها VNF باشد، به آن VNF-FG گفته می‌شود. در صورتی که فرض کنیم همه NF های شکل ۳-۲، VNF هستند می‌توان آن را مطابق شکل ۴-۲ نمایش داد. در این شکل NF2 خود توسط سه VNF پیاده‌سازی شده است.



شکل ۲-۴: گراف VNF-FG متناظر با شکل ۲-۳

مشخص است که گراف VNF-FG صرفاً ارتباطات بین VNFها را مشخص می‌کند ولی ترتیب عبور ترافیک از کارکردها را بیان نمی‌کند. ترتیب عبور ترافیک از کارکردها توسط NFP بیان می‌شود که یکی از اجزای VNF-FG است و هر VNF-FG باید حداقل یک NFP داشته باشد.

۲-۶ موارد کاربرد

در این بخش موارد کاربرد مهم معماری NFV را شرح می‌دهیم. یکی از مهم‌ترین موارد کاربرد ذکر شده برای NFV، مجازی‌سازی تجهیزات CPE است. عموماً تجهیزاتی در مکان کاربران برای اتصال به اینترنت نگهداری می‌شود که شامل دیواره آتش، NAT، مسیریاب و سوئیچ است. در این حالت تنظیمات تجهیزات باید در مکان کاربر صورت بگیرد که هزینه بالایی دارد. با استفاده از مجازی‌سازی این کارکردها و نگهداری آن در سمت ISP، می‌توان هزینه تجهیزات و زمان نگهداری را کاهش داد. این مورد کاربرد در شکل ۲-۵ نمایش داده شده است.

به عنوان مثالی دیگر از موارد کاربرد می‌توان مجازی‌سازی کارکردها در زیرساخت LTE را در نظر گرفت. در شکل ۲-۶ شرایط قبل و بعد از مجازی‌سازی S-GW، P-GW، PCRF و MMF نمایش داده شده است. با مجازی‌سازی این کارکردها علاوه بر استفاده بهتر از منابع، می‌توان تعداد نمونه‌های آن‌ها را مطابق با تعداد کاربران بدون تغییر در زیرساخت افزایش و یا کاهش داد.

۲-۷ زنجیره‌سازی کارکرد سرویس

زنجیره‌سازی کارکردها ایده جدیدی نیست. در حال حاضر اپراتورها برای ارائه سرویس یک زنجیره از کارکردها ایجاد می‌کنند که ترافیک کاربر باید از کارکردها با یک ترتیب



شکل ۲-۵: مجازی‌سازی CPE

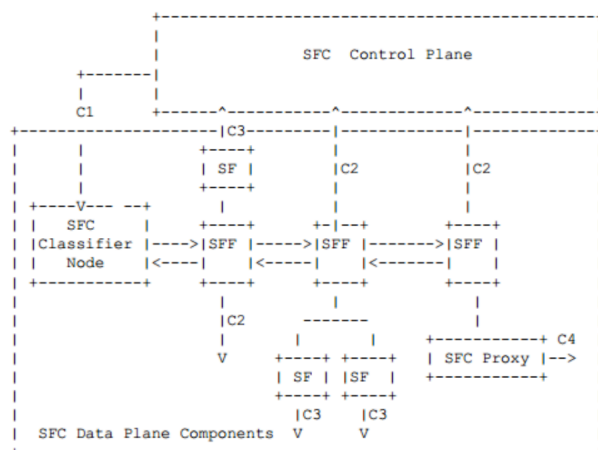


شکل ۲-۶: مجازی‌سازی زیرساخت LTE

مشخص عبور کند. اگرچه همانطور که بیان شد در صورت تغییر در ترتیب کارکردها و یا زنجیره‌ها و یا ایجاد سرویس‌های جدید، نیازمند تغییر مکان فیزیکی کارکردها خواهیم بود که کاری سخت است و خالی از اشکال نیست. معماری زنجیره‌سازی کارکردهای سرویس، اپراتورهای شبکه را قادر می‌سازد که سرویس‌های جدید را به صورت نرم افزاری و پویا و بدون اینکه در سطح سخت افزار تغییری ایجاد کنند، ارائه کنند. در این راستا IETF اسناد متعددی به شرح معماری و اجزای آن پرداخته است. در این بخش به شرح معماری زنجیره‌سازی کارکرد سرویس می‌پردازیم و بخش‌های اصلی آن را بیان می‌کنیم.

۱-۷-۲ اجزای معماری SFC

در این RFC یک سرویس شبکه که توسط اپراتور ارائه می‌شود و از طریق یک یا چند کارکرد سرویس تحویل می‌شود، تعریف شده است. یک کارکرد سرویس، رفتار خاصی (به غیر از جلورانی) با بسته را انجام می‌دهد و می‌تواند در هر یک از لایه‌های مدل OSI فعالیت کند. به یک شبکه یا بخشی از آن که در آن SFC پیاده‌سازی شده است یک دامنه SFC گفته می‌شود. در یک دامنه SFC، معماری SFC مطابق با شکل ۷-۲ پیاده‌سازی می‌شود. معماری SFC توسط RFC 7665 تعریف شده است.



شکل ۷-۲: معماری SFC

به صورت خلاصه اجزای اصلی این معماری عبارتند از:

- زنجیره کارکرد سرویس: یا به صورت خلاصه زنجیره کارکرد یک مجموعه مرتب از کارکردهای سرویس انتزاعی و محدودیت‌های ترتیبی که باید به بسته‌ها، فریم‌ها و یا جریان‌های دریافتی به عنوان نتیجه دسته بندی اعمال شود.
- دسته‌بند: وظیفه دسته‌بندی و انتخاب زنجیره کارکرد برای ترافیک ورودی بر اساس قوانین از پیش تعیین شده را برعهده دارد.
- SFF: وظیفه جلورانی و هدایت ترافیک در دامنه SFC را برعهده دارد.
- کارکرد سرویس (SF): یک کارکرد انتزاعی که مسئول رفتار خاصی با بسته به جز جلورانی است.
- SF Proxy: وظیفه ارتباط با کارکردهای غیر آگاه از کپسول بندی SFC را برعهده دارد.
- صفحه کنترل: وظیفه کنترل و نظارت بر زنجیره‌ها و ایجاد قوانین دسته‌بندی بر روی دسته‌بند را برعهده دارد.

همه این اعضا به صورت منطقی هستند و می‌توانند در شبکه به صورت فیزیکی و یا مجازی در یک یا چندین دستگاه فیزیکی به صورت مشترک با یکدیگر وجود داشته باشند.

۸-۲ جمع‌بندی

در این بخش معماری‌های NFV و SFC به صورت کامل شرح داده شد و اجزای آن‌ها بررسی شد. همانگونه که بیان شد، معماری NFV بر مجازی‌سازی کارکردها تمرکز دارد. یک سرویس در معماری NFV با استفاده از NSD توصیف می‌شود که شامل VNF-FG، VNFها و لینک‌های توصیف کننده ارتباطات بین VNFها است. معماری SFC به ایجاد زنجیره پویا از کارکردها تمرکز دارد. در معماری VNF سرویس‌های مدیریتی و نظارتی مانند VNFM، VNFO و ... نیز تعریف می‌شوند که وظیفه‌ی نظارت و مدیریت چرخه‌ی حیات در این معماری را دارا می‌باشد و تمرکز اصلی این پژوهش بر در نظر گرفتن اهمیت این سرویس‌ها می‌باشد.

یک زنجیره کارکرد توسط یک گراف SFC توصیف می‌شود که به صورت مجموعه مرتب از کارکردها که ترافیک باید با ترتیب مشخصی از آن‌ها عبور کند توصیف می‌شود. این معماری تاکید بر مجازی‌سازی کارکردها ندارد. همچنین برای مسیریابی ترافیک در این معماری نیز می‌توان از سرآیند NSH استفاده کرد. همانگونه که بیان شد در حقیقت این دو معماری مکمل یکدیگر هستند و یک گراف SFC را می‌توان توسط یک VNF-FG معادل نمایش داد.

فصل سوم

کارهای مرتبط

۱-۳ مقدمه

در این بخش تحقیقات مرتبط با استقرار سرویس و تخصیص منابع در معماری‌های NFV و SFC را مورد بررسی قرار می‌دهیم. ابتدا ابعاد مختلف مسائل تحقیقاتی و به خصوص مسئله تخصیص منابع را مورد بررسی قرار می‌دهیم. سپس کارهایی که در زمینه استقرار سرویس و تخصیص منابع مدیریتی انجام شده‌اند را مرور می‌کنیم.

۲-۳ ابعاد مختلف مسائل تحقیقاتی

در این بخش با توجه به معماری‌های SFC و NFV ابعاد مختلف مسائل تحقیقاتی را مورد بررسی قرار می‌دهیم. از آنجایی که موضوع این رساله بر مسئله تخصیص منابع تمرکز دارد بر این بخش تمرکز بیشتری خواهیم داشت. در این تحقیق تخصیص منابع را به صورت اختصاص منابع شبکه، پردازشی، محاسباتی و ذخیره‌سازی به ماشین‌های مجازی اجراکننده کارکردها و اختصاص پهنای باند به لینک‌های مجازی بین کارکردها بر روی شبکه ارتباطی زیرساخت تعریف می‌کنیم. در این راستا باید مشخص شود که ماشین‌های مجازی اجرا کننده کارکردها که به عنوان نمونه ایجاد شده از کارکردها شناخته می‌شوند بر روی چه سرورهایی ایجاد شوند. این فرآیند را نگاشت کارکردها به نمونه‌ها می‌گوییم. همچنین باید مشخص شود چه میزان پهنای باند از چه لینک‌هایی به به لینک‌های مجازی اختصاص یابد. ممکن است پهنای باند یک لینک، بر روی چند لینک و یا چند مسیر در شبکه زیرساخت اختصاص پیدا کند. به این فرایند نیز نگاشت لینک‌های مجازی گفته می‌شود. با توجه به توضیحات گفته شده در ادامه ابعاد مختلف مسائل تحقیقاتی را شرح می‌دهیم.

۱-۲-۳ دیدگاه تعریف مساله

معماری آینده اینترنت بر اساس مدل تجاری IaaS است که نقش ISPها به دو نقش فراهم کننده سرویس (SP) و فراهم کننده زیرساخت (IP)، تبدیل می‌شود فراهم کننده سرویس مسئول ارائه سرویس انتها به انتها به کاربران بر روی زیرساختی است که از سمت IP ارائه می‌شود و مسئولیت مدیریت منابع آن را برعهده دارد. بر اساس این دو نقش، مسائل را از دو جنبه می‌توان دسته‌بندی کرد:

- مسائلی که در آن صرفاً بحث سرویس گرفتن از یک یا چند IP مطرح است. از آنجایی که سرویس گیرنده خود تخصیص منابع را انجام نمی‌دهد، تمرکز این گونه مسئله‌ها بر روی مسائل قیمت‌گذاری در یک بازار NFV خواهد بود.

• مسائلی که در آن‌ها تخصیص بهینه منابع به سرویس نیز مورد توجه است. در این حالت IP از یک مرکز داده متمرکز یا چندین مرکز داده توزیع شده برای ارائه سرویس به کاربران استفاده می‌کند. در این بخش وابسته به سطح انتزاع مسئله، NFVI-PoP را می‌توان یک سرور یا یک مرکز داده در نظر گرفت. طبیعتاً وابسته به سطح انتزاع، همبندی‌های مختلف شبکه ارتباطی را نیز می‌توان در نظر گرفت.

در هر یک از این دیدگاه‌ها می‌توان فرضیاتی را در نظر گرفت که منجر به مسائل متفاوتی می‌شود. به عنوان مثال در دسته اول نحوه قیمت‌گذاری، همکاری و یا عدم همکاری IPها و کاربران را می‌توان مورد مطالعه قرار داد. در مسائل تخصیص منابع هم اگر یک مرکز داده وجود داشته باشد، درباره شبکه زیرساخت می‌توان انواع همبندی‌های Switch centeric و یا Server centric را در نظر گرفت که تاثیر زیادی بر تعریف مسائل دارند. زمانی که چندین مرکز داده توزیع شده از نظر جغرافیایی وجود داشته باشد، مسائل مهمی از جمله نحوه کاهش ترافیک بین مراکز داده مطرح می‌شود. همانگونه که بیان شد تمرکز این تحقیق بر دسته دوم مسائل یعنی تخصیص بهینه منابع به سرویس است.

۳-۳ مرور پژوهش‌ها

در [۶] نویسندگان قصد دارند با در نظر گرفتن محدودیت ظرفیت لینک‌ها و محدودیت پردازشی نودها بیشترین تعداد زنجیره‌ی کارکرد را بپذیرند. برای این کار یک مسالهی ILP طراحی می‌کنند و ثابت می‌کنند که این مساله NP-Hard می‌باشد. با توجه به NP-Hard بودن مساله الگوریتم مکاشفه‌ای MASRN پیشنهاد می‌گردد. این الگوریتم یک الگوریتم حریصانه می‌باشد که براساس منابع سرورها و بار لینک‌ها جایگذاری را انجام می‌دهد. در این مقاله وجود VNFM برای زنجیره‌ها در نظر گرفته نشده است.

در [۲] نویسندگان استفاده از VNFM را مدنظر قرار داده‌اند. در این مقاله فرض شده است که جایگذاری SFCها صورت گرفته است و می‌خواهیم VNFMها را به صورت دوره‌ای با تغییر نگاشت زنجیره‌ها به گونه‌ای بازنگاشت کنیم که با رعایت شدن نیازمندی‌های کارآیی، هزینه‌ی عملیاتی سیستم حداقل شود. مساله مطرح شده به صورت ILP مدل‌سازی می‌شود. این مقاله هزینه‌ی عملیاتی سیستم را تحت چهار عنوان دسته‌بندی می‌کند: هزینه‌ی مدیریت چرخه‌ی زندگی، هزینه‌ی منابع محاسباتی، هزینه‌ی مهاجرت و هزینه‌ی بازنگاشت. در این مقاله فرض می‌شود که هر نمونه از VNFMها می‌تواند به تعداد مشخصی از نمونه‌های VNF سرویس‌دهی کند و این سرویس‌دهی به نوع نمونه وابسته نیست. این مقاله محدودیت‌های پردازشی و ظرفیتی را مدنظر قرار می‌دهد.

در [۸] نویسندگان سه مرحله برای عملیات جایگذاری زنجیره‌های کارکرد سرویس

معرفی می‌کنند: انتخاب، جایگذاری و مسیریابی. در این مقاله فرض می‌شود برای هر نوع VNF چند مدل مختلف با مصرف منابع مختلف وجود دارند که می‌توان از آن‌ها نمونه ساخت. در این مرحله مشخص می‌شود از کدام مدل نمونه‌سازی صورت می‌گیرد. این مقاله جایگذاری یک SFC را مدل‌سازی می‌کند، در این مقاله فرض می‌شود جریان ورودی و خروجی از هر نمونه برابر بوده و در واقع VNF تغییری بر روی ترافیک ایجاد نمی‌کند. در مدل‌سازی این مقاله که به صورت ILP می‌باشد هدف کاهش هزینه در جایگذاری SFC داده شده می‌باشد. با در نظر گرفتن مدل‌های مختلف برای VNF‌ها در این مقاله در صورتی که نیاز به پردازش ترافیک زیادی باشد، چند نمونه از یک نوع VNF ساخته می‌شود و ترافیک بین آن‌ها تقسیم می‌شود.

در [۱۴] نویسندگان برای اولین بار مساله‌ی Traffic Steering را با در نظر گرفتن QoS و Reliability فرمول‌بندی کرده‌اند. این مقاله کاربرد NFV را در شبکه‌های موبایل مدنظر قرار داده است. در این مقاله مساله به صورت Link-Path مدل‌سازی شده است و فرض شده است که مسیرهای ممکن برای جایگذاری کلاس‌های ترافیکی از پیش تعیین شده‌اند. در این مقاله منظور از کیفیت سرویس تاخیر و گذردهی کلاس‌های ترافیکی می‌باشد و برای فراهم آوردن قابلیت اطمینان فرض می‌شود که خرابی‌ها به صورت دلخواه بوده و در صورت خرابی بخشی از پهنای باند از دست می‌رود.

در [۹] نویسندگان مساله‌ی جایگذاری و مسیریابی زنجیره‌های کارکرد سرویس را به صورت توامان مدل‌سازی می‌کنند، در این مساله نویسندگان تاثیر دو پارامتر Coordination Effect و Traffic-Change Effect را نیز مدنظر قرار داده‌اند. زمانی که چند VM در پیاده‌سازی یک کارکرد شبکه استفاده می‌شوند نیاز است که بین این ماشین‌های مجازی هماهنگی صورت بگیرد. برای این هماهنگی ارتباطاتی صورت می‌گیرد که دارای سربار بوده و به این سربار Coordination Effect می‌گویند. هر کارکرد شبکه می‌تواند روی ترافیک ورودی خود تاثیر گذاشته و نرخ آن را تغییر دهد که این موضوع را با Traffic-Change Effect بیان می‌کنند.

در [۵] نویسندگان قصد دارند به صورت قطعی کیفیت سرویس را گارانتی نمایند. این مقاله پیاده‌سازی NFV را با استفاده از SDN هدف قرار می‌دهد و برای محاسبه‌ی تاخیر، تاخیر پیام‌های کنترلی SDN و تاخیر جابجایی بسته‌ها را در نظر می‌گیرد. برای پیشنهاد یک راه‌حل قطعی از Network Calculus استفاده می‌شود که شرایط مرزی را بررسی می‌کند. این شرایط مرزی برای پیام‌های کنترلی محاسبه شده و از آن تاخیر مورد نظر در جابجایی بسته‌ها بدست می‌آید که با استفاده از آن یک مساله‌ی بهینه‌سازی با هدف رعایت تاخیر بدست آمده حاصل می‌شود.

در [۱۱] نویسندگان پیاده‌سازی NFV با SDN را هدف قرار داده‌اند و جایگذاری middle

boxها با هدف توزیع بار را فرمول‌بندی کرده‌اند. در واقع middle boxها در این مقاله به صورت مجازی بوده و همان کارکردهای مجازی شبکه می‌باشند. مدل‌سازی صورت گرفته به صورت node link صورت پذیرفته است. هدف مساله مسیریابی چند مسیره برای تقاضا به صورتی است که در آن link load ratio برای تمام لینک‌ها کمینه شود. این مقاله تغییر ترافیک توسط کارکردها را نیز مدنظر قرار داده است.

در [۱۰] مساله‌ی جایگذاری زنجیره‌های کاکرد سرویس با دو هدف کاهش مصرف انرژی و افزایش نرخ جریان پذیرفته شده مدل‌سازی می‌شود. این مدل‌سازی با توجه به معماری IETF SFC صورت پذیرفته است. در مدل‌سازی این مقاله جزئیات زیادی مورد توجه قرار گرفته است که این امر باعث پیچیده شدن فرمول‌بندی شده است.

در [۷] نویسندگان ابتدا مساله‌ی جایگذاری و مسیریابی VNFها را در اوج ترافیک حل می‌کنند. در ادامه آن‌ها فرض می‌کنند که ترافیک به صورت دوره‌ای-ثابت می‌باشد به این معنا که ترافیک در تعداد متناهی بازه‌ی زمانی تعریف شده و تکرار می‌شود. با این فرض در ادامه، مقاله مساله‌ی دیگری مبنی بر مهاجرت نمونه‌ها با توجه به تغییر ترافیک را مطرح می‌کند. در این مهاجرت‌ها مقاله از توان مصرفی در مهاجرت صرف نظر کرده و تلاش می‌کند جریمه‌ای که بابت قطعی سرویس پرداخت می‌شود و توان مصرفی کل سیستم را بهینه کند.

در [۱۳] نویسندگان مساله‌ی توزیع بار در NFV را بررسی می‌کنند، آن‌ها در این مساله ویژگی‌های پایه‌ای NFV در کنار استفاده از روش ECMP مدنظر قرار می‌دهند. در روش ECMP بار بین مبدا و مقصد به صورت یکسان بین تمام مسیرها تقسیم می‌گردد. در این مساله تعدادی تقاضا در نظر گرفته می‌شود که کوتاهترین مسیرها بین مبدا و مقصد آن‌ها مشخص است و در نهایت بار در این مسیرها توزیع شده و کارکردها شبکه‌ای نیز در این مسیرها مستقر می‌شوند.

همانطور که در جدول ۱-۳ دیده می‌شود، مساله‌ی تخصیص منابع مدیریتی در پذیرش زنجیره‌های کارکرد مورد بررسی قرار نگرفته است و در پژوهش حاضر قصد داریم این مورد را بررسی نماییم.

نزدیک‌ترین پژوهش به آنچه در این رساله انجام شده است، [۲] می‌باشد. این پژوهش قصد دارد با فرض مشخص بودن نگاشت زنجیره‌ها و NFVO نگاشت VNFM را انجام دهد. این پژوهش فرض می‌کند که در طی بازه‌های زمانی مختلف نگاشت زنجیره‌ها تغییر می‌کند و این الگوریتم قصد دارد با کمترین هزینه‌ی عملیاتی نگاشت دوباره VNMFها را انجام دهد. در این نگاشت پارامترهایی مانند تاخیر، هزینه‌ی پهنای باند، هزینه‌ی منابع مدیریتی را مدنظر قرار می‌گیرند. در بازنگاشت VNFMها لینک ارتباطی آن‌ها با VNFO نیز مدنظر است.

جدول ۳-۱: مقایسه مقالات پذیرش زنجیره‌های کارکرد سرویس

منبع	مقایسه تخصیص یافته	محدودیت ظرفیت پردازشی نمونه	برخط یا برون خط	منبع	مقایسه تخصیص یافته	محدودیت ظرفیت پردازشی نمونه	برخط یا برون خط
#	other	MEM	BW	CPU	دارد	ندارد	برخط یا برون خط
[۶]	—	—	✓	✓	—	✓	—
[۸]	—	—	✓	✓	✓	—	—
[۹]	—	—	✓	✓	✓	—	—
پژوهش حاضر	—	✓	✓	✓	✓	—	—
منبع	نگاشت کارکرد و لینک	انتساب کارکرد	اشتراک نمونه	تخصیص VNFM	منبع	نگاشت کارکرد و لینک	انتساب کارکرد
#	کارکرد	لینک	یک نمونه	چند نمونه	دارد	ندارد	دارد
[۶]	✓	✓	✓	—	—	—	—
[۸]	✓	✓	—	✓	—	—	—
[۹]	✓	✓	—	✓	—	—	—
پژوهش حاضر	✓	✓	✓	—	—	—	—

پژوهش [۲] برای VNFM ها نوع متصور شده و فرض می کند هر نوع از VNFM ها می تواند از تعداد مشخصی نمونه VNF پشتیبانی کند، این فرض مشابه فرضی می باشد که در پژوهش حاضر نیز صورت گرفته است.

تفاوت های اصلی پژوهش حاضر با پژوهش [۲] به شرح زیر می باشد:

- پژوهش حاضر هزینه ی گواهی VNFM ها را مدنظر قرار داده است.
- پژوهش حاضر یک مساله پذیرش برای زنجیره ها را حل می کند که در این پذیرش در کنار نگاشت زنجیره ها منابع مدیریتی آن ها را نیز نگاشت می کند.
- پژوهش حاضر VNFO را در نظر نمی گیرد و مساله را برای یک دیتاسنتر توزیع نشده مطرح می کند.
- پژوهش حاضر ممکن است در جهت افزایش سود زنجیره ای را نپذیرد چرا که ممکن است نگاشت VNFM برای آن سربار زیادی داشته باشد و این در حالی است که پژوهش [۲] چنین فرضی نداشته و برای تمامی زنجیره ها منابع مدیریتی را نگاشت می کند.

در نهایت با توجه به متفاوت بودن تابع های هدف این دو پژوهش نمی توان آن ها را با یکدیگر مقایسه کرد.

۳-۴ جمع‌بندی

در این فصل انواع مسائل مطرح در حوزه‌ی NFV و SFC به صورت خلاصه مرور گشت. همانطور که پیشتر بیان شده بود مساله‌ی پیشنهادی یک مساله‌ی جایگذاری یا نگاشت سرویس می‌باشد. در ادامه پژوهش‌های این حوزه مرور شده و تفاوت‌های آن‌ها با پژوهش حاضر ذکر شد. در آخر پژوهش حاضر سعی دارد برای اولین بار نگاشت سرویس‌ها را با در نظر گرفتن نیاز آن‌ها به منابع مدیریتی مدنظر قرار دهد.

فصل چہارم

تعریف مسالہ

۴-۱ مقدمه

یکی از نیازمندی‌هایی که در [۱] بیان شده است، نیاز هر یک از نمونه‌های یک زنجیره‌ی کارکرد سرویس به یک VNFM می‌باشد. قصد این رساله تعریف مساله‌ای جامع و واقعی است که بتواند این موضوع را نیز در برگیرد چرا که این نیازمندی برای مراکز داده بسیار مهم بوده و در صورت در نظر نگرفتن آن ممکن است به کیفیت سرویس آن‌ها لطمه بخورد. مساله به صورت توامان جایگذاری زنجیره‌ها و VNFM برای آن‌ها را مدنظر قرار می‌دهد و اجازه می‌دهد که سیاست‌های زیر به مساله اعمال شود:

- برخی از سرورهای فیزیکی ممکن نباید توسط سرورهای فیزیکی مشخصی مدیریت شوند. مثلاً به علت فاصله زیادی که با یکدیگر دارند یا به علت مسائل امنیتی
- برخی از کارکردهای مجازی نیاز به سخت‌افزارهای خاصی دارند که ممکن است تنها بتوانند بر روی سرورهای فیزیکی مشخصی قرار گیرند.

در کنار این سیاست‌ها سعی شده است منابع مدیریتی به صورت کامل مدل‌سازی شوند. در دنیای امروز برای استفاده از VNFM‌های تجاری نیاز است که هزینه‌ای پرداخت شود و هر یک از نمونه‌های تهیه شده می‌تواند تعداد مشخصی از کارکردها را مدیریت کند. در کنار این امر، VNFM‌ها منابع پردازی مصرف می‌کنند و نیاز دارند که کارکردهایی که مدیریت می‌کنند در شعاع خاصی از آن‌ها قرار داشته باشند. تمامی این موارد در مساله‌ی طرح شده مدنظر می‌باشند. در این فصل فرمول‌بندی مساله و تنظیمات آن را مرور می‌کنیم و در آخر یک مثال از این مساله را می‌بینیم.

۴-۲ مساله

بیشینه کردن سود حاصل از پذیرفتن تقاضای زنجیره کارکرد سرویس با در نظر گرفتن انتساب هر نمونه کارکرد مجازی شبکه به یک VNFM. همانطور که در مستند [۱] نیز آمده است، نیاز است که هر یک نمونه‌های کارکردهای مجازی شبکه توسط حداقل یک VNFM مدیریت شوند. در این مساله قصد داریم مساله پذیرش تقاضاهای زنجیره‌های کارکرد سرویس را با نظر گرفتن این نیازمندی در کنار نیازمندی‌های پردازی و پهنای‌باند هر یک از تقاضاها حل کنیم.

در ادامه به صورت خلاصه شرایط مساله را بررسی می‌کنیم:

- توپولوژی زیرساخت شامل پهنای باند لینک‌ها و ظرفیت NFVI-PoP^۱ها^۱ موجود است.

^۱NFVI Point of Presence

- n تقاضای زنجیره کارکرد سرویس به صورت کامل و از پیش مشخص شده داریم.
- هر تقاضا شامل نوع و تعداد نمونه‌های مجازی، پنهان باند لینک‌های مجازی و توپولوژی نمونه‌های مجازی می‌باشد.
- F نوع کارکرد مجازی شبکه تعریف شده است که هر یک مقدار مشخصی از حافظه و توان پردازشی را مصرف می‌کنند.
- تعداد پردازنده‌هایی که به هر نمونه تخصیص می‌یابد با توجه به ترافیک ورودی نمونه مشخص می‌شود. این امر توسط اپراتور در زمان تعریف مساله ورودی صورت می‌گیرد.
- نمونه‌ها بین زنجیره‌ها به اشتراک گذاشته نمی‌شوند.
- محدودیت ظرفیت لینک‌ها
- محدودیت توان پردازش سرورهای فیزیکی با توجه به میزان حافظه و تعداد پردازنده‌ها
- برای مدیریت یکدست و آسان‌تر زنجیره‌ها و در عین حال جمع‌آوری راحت‌تر خطاها، برای هر زنجیره یک VNFM فیزیکی تخصیص می‌دهیم.
- VNFM‌ها می‌توانند بین زنجیره به اشتراک گذاشته شوند.
- هر نمونه از VNFM‌ها می‌تواند تعداد مشخصی از نمونه‌های کارکرد مجازی شبکه را سرویس دهد.
- برای ارتباط میان هر نمونه از VNFM‌ها و VNF‌ها پهنای باند مشخصی رزرو می‌گردد.
- در صورتی که NFVI-PoP بتواند از VNFM پشتیبانی نماید می‌توان به هر تعداد که ظرفیت آن اجازه می‌دهد بر روی آن VNFM مستقر کرد.
- هر نمونه از VNFM جهت استفاده نیاز به تهیه جواز^۲ دارد.
- توپولوژی می‌تواند دارای تعدادی گرهی ورودی^۳ و خروجی^۴ باشد.
- هر زنجیره می‌تواند دارای تعدادی نقطه‌ی ورودی و خروجی باشد که می‌بایست بر روی گره‌های ورودی و خروجی نگاشته شوند.
- اگر جایگذاری VNFM‌ها به صورت غیر برنامه‌ریزی شده صورت بپذیرد ممکن است به تاخیرهای غیرقابل تحمل منجر شده و به این ترتیب تاثیر منفی بر روی کارایی سیستم داشته باشد.

^۲ license

^۳ ingress

^۴ egress

یکی از وظایف VNFM ها جمع‌آوری پیام‌های خطا می‌باشد. برای این امر نیاز است که پهنای باند کوچک اما اختصاصی به VNFM ها تخصیص داده شود بنابراین نمی‌توان جایگذاری آن‌ها را با روش‌های سابق و مانند سایر کارکردهای مجازی شبکه فرض کرد. از آنجایی که VNFM ها نیاز به مجوز دارند می‌توان با به اشتراک گذاشتن آن‌ها در هزینه‌های سیستم صرفه‌جویی کرد. در نظر گرفتن VNFM همراه با VNF ها مساله‌ی جدیدی است.

۳-۴ مدل سیستم

۱-۳-۴ شبکه‌ی زیرساخت

شبکه‌ی زیرساخت با گراف وزن دار $G_S^{PN} = (V_S^{PN}, E_S^{PN})$ نشان داده می‌شود که در آن V_S^{PN} نشان‌دهنده‌ی مجموعه‌ی گره‌های زیرساخت و E_S^{PN} نشان‌دهنده‌ی مجموعه‌ی یال‌های شبکه زیرساخت است.

- تمامی گره‌ها مقداری حافظه و تعدادی هسته دارند.
- برخی از گره‌ها ممکن است نتوانند توسط گره‌های مشخصی مدیریت شوند.
- برخی از گره‌ها ممکن است توانایی پشتیبانی از سرویس‌های مجازی را نداشته باشند.
- گره‌های مشخصی در شبکه وجود دارند که می‌توانند در نقش گره‌ی ورودی و خروجی عمل کنند.
- یال‌های شبکه همگی دارای ظرفیت مشخصی می‌باشند.

۲-۳-۴ منابع مدیریتی

هر یک از VNFM ها نیاز به یک گواهی دارند که با مصرف مقدار مشخصی از حافظه و هسته‌های پردازنده می‌توانند تعداد مشخصی از نمونه‌ها را سرویس دهند.

۳-۳-۴ انواع

هر یک از انواع کارکردهای مجازی شبکه نیاز به مقدار مشخصی از حافظه و هسته‌های پردازنده دارند و ممکن است نیازمند مدیریت شدن باشند یا این نیاز را نداشته باشند. برخی از کارکردها نیاز دارند بر روی نودهای ورودی یا خروجی قرار گیرند.

۴-۳-۴ زنجیره‌ها

زنجیره‌ی i ام با گراف وزن دار $G_i^{SFC} = (V_{i,F}^{SFC}, E_i^{SFC})$ نشان داده می‌شوند که در آن $V_{i,F}^{SFC}$ نشان‌دهنده‌ی مجموعه‌ی گره‌های زنجیره و E_i^{SFC} یال‌های زنجیره می‌باشد. تمامی گره‌ها دارای یک نوع مشخص می‌باشند و تمامی یال‌ها ظرفیت مشخصی دارند.

۴-۴ فرمول‌بندی

هدف اصلی مساله پذیرش بیشترین تعداد تقاضا می‌باشد. در اینجا فرض می‌کنیم پذیرش هر تقاضا سودی منحصر به فرد و هزینه‌ای برای تهیه گواهی VNFM در بر خواهد داشت. بنابراین تابع هدف به شکل زیر می‌باشد:

$$\max \sum_{h=1}^T c_h x_h - \sum_{w \in V_s^{PN}} licenseFee.w \quad (4-1)$$

جدول ۴-۱: پارامترهای مساله

$memory(k)$	required RAM of VNF instance with type k in GB
$core(k)$	required CPU cores of VNF instance with type k
$memory$	required RAM of VNFM in GB
$core$	required CPU cores of VNFM
$capacity$	maximum number of VNF instances that VNFM can handle
$len(h)$	number of VNF instances in h th SFC request
$type(v, k)$	assuming the value 1 if the VNF instance v has type k
$bandwidth(u, v)$	required bandwidth in link from VNF instance u to v
$\hat{bandwidth}$	required bandwidth in management link
$radius$	maximum neighborhood distance for instance management
$licenseFee$	VNFM license fee that must pay for each VNFM
$vnfSupport(w)$	assuming the value 1 if the physical server w can support VNF instances
$isManageable(k)$	assuming the value 1 if the type k needs a manager
$notManageableBy(w1, w2)$	assuming the value 1 if the physical server $w1$ cannot manage by physical server $w2$

برای هر نود اندازه‌ی مشخصی از حافظه RAM در نظر گرفته می‌شود که هر نمونه‌ی کارکرد با توجه به نوع آن مقدار مشخصی از این حافظه را مصرف می‌کند. VNFM نیز مقدار مشخصی از حافظه را مصرف می‌کند.

جدول ۴-۲: متغیرهای تصمیم‌گیری مساله (قسمت اول)

x_h	binary variable assuming the value 1 if the h th SFC request is accepted; otherwise its value is zero
y_{wk}	the number of VNF instances of type k that are used in server $w \in V_s^{PN}$
z_{vw}^k	binary variable assuming the value 1 if the VNF node $v \in \cup_{i=1}^T V_{i,F}^{SFC}$ is served by the VNF instance of type k in the server $w \in V_s^{PN}$
\bar{y}_w	the number of VNFMs (each vnf has its capacity and license fee) that are used in server $w \in V_s^{PN}$
\bar{z}_{hw}	binary variable assuming the value 1 if h th SFC is assigned to VNFM on server $w \in V_s^{PN}$

Node Memory Constraint:

$$\sum_{k=1}^F y_{wk} \text{memory}(k) + \bar{y}_w \text{memory} \leq N_{ram}^{PN}(w) \quad \forall w \in V_s^{PN} \quad (4-2)$$

برای هر نود تعداد مشخصی از هسته‌های پردازنده در نظر گرفته می‌شود که هر نمونه‌ی کارکرد با توجه به نوع آن مقدار مشخصی از این تعداد را مصرف می‌کند. VNFM نیز مقدار مشخصی از تعداد هسته‌های پردازنده را مصرف می‌کند.

Node CPU Constraint:

$$\sum_{k=1}^F y_{wk} \text{core}(k) + \bar{y}_w \text{core} \leq N_{core}^{PN}(w) \quad \forall w \in V_s^{PN} \quad (4-3)$$

اگر VNF، v توسط VNF instance نوع k روی سرور w سرویس شود می‌بایست VNF instance نوع k روی سرور w فعال شود. توجه شود که اشتراک‌گذاری VNFها پشتیبانی نمی‌گردد.

Service Place Constraint:

$$\sum_{v \in \cup_{i=1}^T V_{i,F}^{SFC}} z_{vw}^k \leq y_{wk} \quad \forall w \in V_s^{PN}, \forall k \in [1, \dots, F] \quad (4-4)$$

اگر تقاضای h ام پذیرفته شده باشد می‌بایست تمام VNF nodeهای آن سرویس شده باشند. یک VNF حداکثر یک بار سرویس داده شود.

Service Constraint:

$$x_h = \sum_{k=1}^F \sum_{w \in V_s^{PN}} z_{vw}^k \quad \forall v \in V_{h,F}^{SFC}, \forall h \in [1, \dots, T] \quad (4-5)$$

اگر تقاضای h پذیرفته شده باشد می‌بایست توسط یک VNFM سرویس شده باشد.
Manage Constraint:

$$x_h = \sum_{w \in V_s^{PN}} \bar{z}_{hw} \quad \forall h \in [1, \dots, T] \quad (4-6)$$

محدودیت ظرفیت سرویس‌دهی VNFM. این محدودیت براساس تعداد ماشین‌های مجازی که هر VNFM سرویس می‌دهد تعیین شده است. در نظر داشته باشید که ممکن است برخی از انواع VNFها نیازی به مدیریت شدن نداشته باشند.

Manage Capacity Constraint & Manage Place Constraint:

$$\sum_{i=1}^T \bar{z}_{iw} \cdot (\text{len}(i) - \sum_{v \in V_{i,F}^{SFC}} \sum_{k \in [1, \dots, F]} \text{type}(v, k) \cdot \text{isManageable}(k)) \leq \text{capacity} \cdot \bar{y}_w \quad \forall w \in V_s^{PN} \quad (4-7)$$

اگر VNF، v توسط instance نوع k روی سرور w سرویس شود می‌بایست خود از نوع k باشد.

Type Constraint:

$$z_{vw}^k \leq \text{type}(v, k) \quad \forall w \in V_s^{PN}, \forall k \in [1, \dots, F], \forall v \in \cup_{i=1}^T V_{i,F}^{SFC} \quad (4-8)$$

در صورتی که سرور w توانایی اجرای نمونه‌های VNF را نداشته باشد نباید نمونه‌ای روی آن قرار گیرد.

VNF support constraint

$$\sum_{k \in [1, \dots, F]} y_{wk} \leq M \cdot \text{vnfSupport}(w) \quad w \in V_s^{PN} \quad (4-9)$$

برخی از سرورهای نمی‌توانند توسط سرورهای مشخصی مدیریت شوند. این ویژگی به ادمین شبکه امکان مدیریت بیشتری می‌دهد و او می‌تواند با دست باز تمامی سیاست‌های مورد نظرش را اعمال نماید.

Manager to node support constraint

$$1 - z_{vw_1}^k + \bar{z}_{hw_2} = 0 \quad \forall w_1 \in V_s^{PN} \forall w_2 \in V_s^{PN} \text{notManagableBy}(w_1, w_2) = 1 \\ \forall h \in [1, \dots, T], \forall v \in V_{h,F}^{SFC}, \forall k \in [1, \dots, T] \quad (4-10)$$

جدول ۴-۳: متغیرهای تصمیم‌گیری مساله (قسمت دوم)

$\tau_{ij}^{(u,v)}$	binary variable assuming the value 1 if the virtual link (u, v) is routed on the physical network link (i, j)
$\bar{\tau}_{ij}^v$	binary variable assuming the value 1 if the managemnt of VNF node v is routed on the physical network link (i, j)

محدودیت زیر بقای جریان در لینک‌های مورد تقاضای کاربر را تضمین می‌کند.

Flow Conservation:

$$\sum_{(i,j) \in E^{PN}} \tau_{ij}^{(u,v)} - \sum_{(j,i) \in E^{PN}} \tau_{ji}^{(u,v)} = \sum_{k=1}^F z_{ui}^k - \sum_{k=1}^F z_{vi}^k$$

$$\forall i \in V_S^{PN}, (u, v) \in E_h^{SFC}, h \in [1, \dots, T] \quad (4-11)$$

محدودیت زیر بقای جریان در لینک‌های مدیریتی را تضمین می‌کند.

Management flow Conservation:

$$\sum_{(i,j) \in E^{PN}} \bar{\tau}_{ij}^v - \sum_{(j,i) \in E^{PN}} \bar{\tau}_{ji}^v = \sum_{k=1}^F z_{vi}^k - \bar{z}_{hi}$$

$$\forall i \in V_S^{PN}, v \in V_{h,F}^{SFC}, h \in [1, \dots, T] \quad (4-12)$$

محدودیت ظرفیت لینک‌ها. در نظر داشته باشید که لینک‌های مدیریتی نیز مقدار کمی از پهنای باند را به صورت رزرو شده احتیاج دارند.

Link Bandwidth Constraint:

$$\sum_{v \in \cup_{i=1}^T V_{i,F}^{SFC}} \bar{\tau}_{ij}^v \cdot \text{bandwidth} + \sum_{(u,v) \in \cup_{i=1}^T E_i^{SFC}} \tau_{ij}^{(u,v)} \cdot \text{bandwidth}(u, v) \leq C_{ij}$$

$$\forall (i, j) \in E^{PN} \quad (4-13)$$

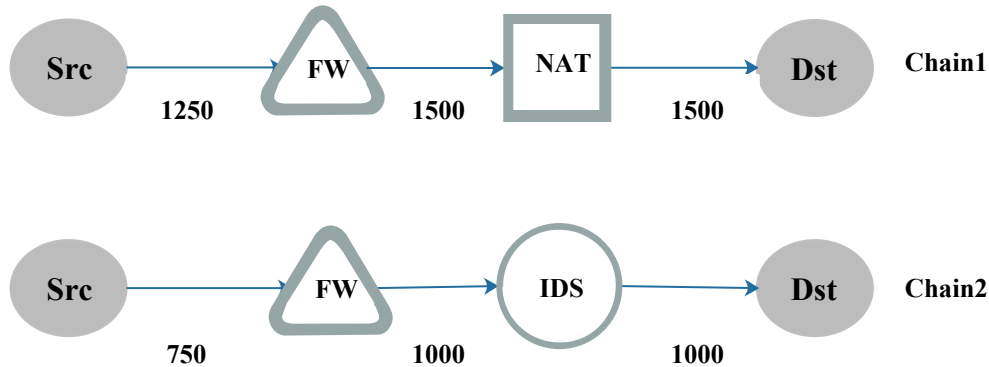
شعاع همسایگی تضمین می‌کند که زمان سرویس‌دهی توسط VNFMها در یک بازه مشخص (از نظر تعداد گام) خواهد بود.

Radius Constraint

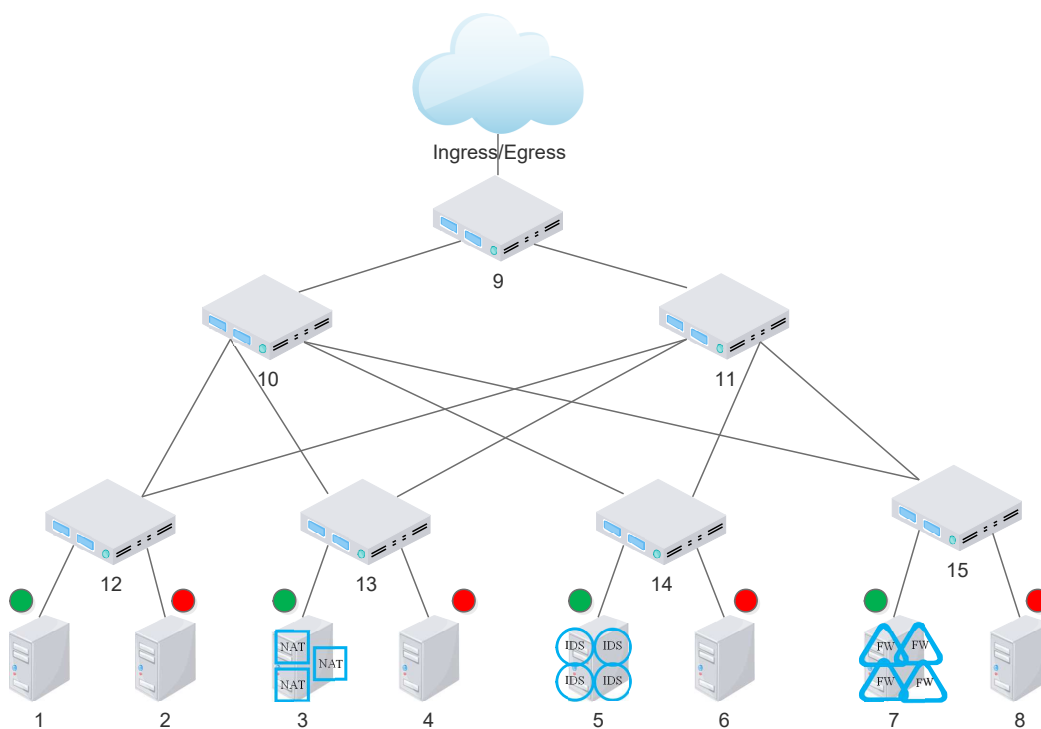
$$\sum_{(i,j) \in E^{PN}} \bar{\tau}_{ij}^v \leq \text{radius} \quad \forall v \in \cup_{i=1}^T V_{i,F}^{SFC} \quad (4-14)$$

۵-۴ مساله‌ی نمونه

قصد داریم زنجیره‌های زیر شکل ۱-۴ را در توپولوژی شکل ۲-۴ جایگذاری نماییم.



شکل ۱-۴: زنجیره‌های مساله‌ی نمونه



شکل ۲-۴: توپولوژی زیرساخت مساله‌ی نمونه

نیازمندی‌های نمونه‌ها در جدول ۴-۴ آمده است و منابع سرورها نیز در جدول ۵-۴ جمع‌آوری شده است.

نمونه‌ها تنها می‌توانند روی سرورهای ۱، ۳، ۵ و ۷ قرار گیرند. مدیریت سرورهای ۱ و ۳ تنها می‌تواند روی سرورهای ۲ و ۴ انجام شود، مدیریت سرور ۵ تنها می‌تواند روی سرورهای ۴ و ۶ صورت گیرد و در نهایت مدیریت سرور ۷ تنها می‌تواند روی سرورهای ۶

جدول ۴-۴: نیازمندی نمونه‌های مساله‌ی نمونه

Spec/VNF	vFW	vNAT	vIDS
CPU (vCore)	2	2	2
Memory (GB)	2	4	2

جدول ۴-۵: مشخصات سرورهای زیرساخت مساله‌ی نمونه

	Server 1,2,7,8	Servers 3,4,5,6
Installed vCPU	144	72
Installed Memory (GB)	1408	288
Link (Gbps)	40	40

و ۸ صورت پذیرد. هر VNFM تنها می‌تواند ۵ نمونه را پشتیبانی کند و برای این امر نیاز به ۴ گیگابایت حافظه و ۲ هسته‌ی پردازشی دارد. تمامی لینک‌های فیزیکی ظرفیت ۴۰ گیگابیت بر ثانیه را دارا می‌باشند. این مثال به صورت کامل بر روی چهارچوب این مساله قابل پیاده‌سازی می‌باشد. نتایج در جدول ۴-۶ خلاصه شده‌اند.

جدول ۴-۶: نتایج مساله‌ی نمونه

	Src	Node-0	Node-1	Dst	VNFM
Chain 0	Switch-9	Server-7	Server-5	Switch-9	Server-6
Chain 1	Switch-9	Server-3	Server-3	Switch-9	Server-4

در نهایت جایگذاری لینک‌ها به شرح زیر می‌باشد:

Chain 0 link 0 (0 - 1) is on switch-9 - switch-11

Chain 0 link 0 (0 - 1) is on switch-11 - switch-15

Chain 0 link 0 (0 - 1) is on switch-15 - server-7

Chain 0 link 1 (1 - 2) is on server-7 - switch-15

Chain 0 link 1 (1 - 2) is on switch-15 - switch-11

Chain 0 link 1 (1 - 2) is on switch-11 - switch-9

Chain 0 link 1 (1 - 2) is on switch-9 - switch-10

Chain 0 link 1 (1 - 2) is on switch-10 - switch-14

Chain 0 link 1 (1 - 2) is on switch-14 - server-5

Chain 0 link 2 (2 - 3) is on server-5 - switch-14

Chain 0 link 2 (2 - 3) is on switch-14 - switch-11

Chain 0 link 2 (2 - 3) is on switch-11 - switch-9

Chain 0 node 1 manager is on server-7 - switch-15

Chain 0 node 1 manager is on switch-15 - switch-11

Chain 0 node 1 manager is on switch-11 - switch-14

Chain 0 node 1 manager is on switch-14 - server-6

Chain 0 node 2 manager is on server-5 - switch-14

Chain 0 node 2 manager is on switch-14 - server-6

Chain 1 link 0 (0 - 1) is on switch-9 - switch-10

Chain 1 link 0 (0 - 1) is on switch-10 - switch-14

Chain 1 link 0 (0 - 1) is on switch-14 - switch-11

Chain 1 link 0 (0 - 1) is on switch-11 - switch-13

Chain 1 link 0 (0 - 1) is on switch-13 - server-3

Chain 1 link 1 (1 - 2) is on switch-10 - switch-14

Chain 1 link 1 (1 - 2) is on switch-14 - switch-11

Chain 1 link 1 (1 - 2) is on switch-11 - switch-15

Chain 1 link 1 (1 - 2) is on switch-15 - switch-10

Chain 1 link 2 (2 - 3) is on server-3 - switch-13

Chain 1 link 2 (2 - 3) is on switch-13 - switch-10

Chain 1 link 2 (2 - 3) is on switch-10 - switch-15

Chain 1 link 2 (2 - 3) is on switch-15 - switch-11

Chain 1 link 2 (2 - 3) is on switch-11 - switch-14

Chain 1 link 2 (2 - 3) is on switch-14 - switch-10

Chain 1 link 2 (2 - 3) is on switch-10 - switch-9

Chain 1 node 1 manager is on server-3 - switch-13

Chain 1 node 1 manager is on switch-13 - server-4

Chain 1 node 2 manager is on server-3 - switch-13

Chain 1 node 2 manager is on switch-13 - server-4

۴-۶ جمع‌بندی

در این بخش مساله‌ی اصلی بیان شده و شرایط آن شرح داده شد. فرمول‌بندی مساله در قالب برنامه‌ریزی خطی صحیح مرور شد و در نهایت برای نمایش بهتر مساله یک مثال زده شد. همانطور که پیشتر هم بیان شده بود، مساله قابلیت تنظیم زیادی داشته و این یکی از نوآوری‌های اصلی این رساله می‌باشد. هدف اصلی مساله نزدیک بودن به واقعیت است و تلاش دارد تمامی سیاست‌های یک مرکز داده در مساله قابل تنظیم باشند.

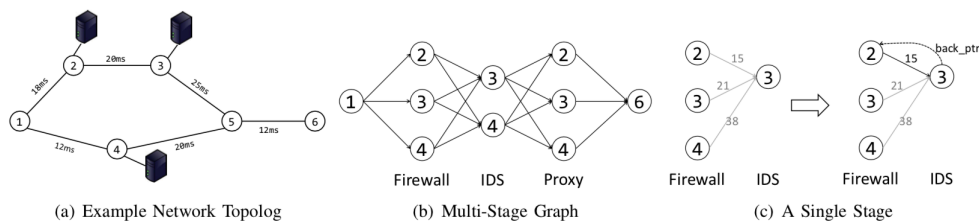
فصل پنجم

راه حل پیشنهادی

مساله‌ی بیان شده به صورت ILP مدل‌سازی می‌شود. در [۶] مساله‌ی جایگذاری SFCها با هدف حداکثرسازی تعداد درخواست‌های پذیرفته شده به صورت ILP مدل‌سازی شده و اثبات شده است که مساله‌ی حاضر NP-Hard می‌باشد. مساله‌ای که در اینجا مدل‌سازی می‌شود از آن مساله پیچیده‌تر می‌باشد زیرا در نظر گرفتن VNFMها را نیز شامل می‌شود. برای این مساله می‌توان یک راه حل مکاشفه‌ای با زمان چند جمله‌ای پیشنهاد داد.

۵-۱ الگوریتم مکاشفه‌ای

مساله از دو قسمت تشکیل شده است. قسمت اول مساله‌ی جایگذاری لینک‌ها و نمونه‌ها می‌باشد و قسمت دوم جایگذاری VNFM برای زنجیره است. برای قسمت اول راه‌حل‌های مکاشفه‌ای زیادی ارائه شده است که ما در اینجا از ایده‌ی راه حل [۴] استفاده می‌کنیم. در این راه حل برای قرارگیری هر زنجیره یک گراف چند گامی شکل می‌گیرد. هر گام این گراف نماینده یک نمونه از زنجیره است که می‌بایست قرار گیرد. در نظر داشته باشید که مساله‌ی اصلی و راه حل بهینه می‌توانند روی ورودی مشابه با VNF-FG فعالیت کنند و این در حالی است که راه حل پیشنهادی تنها می‌تواند پذیرای زنجیره‌ها باشد. در هر گام از این گراف مجموعه‌ای از نودهای فیزیکی امکان‌پذیر شکل می‌گیرد. با توجه به وضعیت مسیریابی این مجموعه با مجموعه بعدی، نود فیزیکی برای نمونه مورد نظر از زنجیره انتخاب می‌شود.



شکل ۵-۱: مدل‌سازی با گراف چندگامی [۴]

منظور از وضعیت مسیریابی به شرح زیر است. برای هر یک از گام‌ها از الگوریتم جستجوی اول سطح یا BFS استفاده می‌کنیم و به این ترتیب مسیرهای فیزیکی که می‌توان از آن‌ها برای جایابی لینک مجازی استفاده کرد پیدا می‌کنیم. از این بین گره‌ای که مسیرهای فیزیکی امکان‌پذیر بیشتری دارد انتخاب می‌گردد. با این روش مجموعه امکان‌پذیر گام بعدی بزرگتر می‌شود و امکان حذف زنجیره به دلیل نبود مسیر فیزیکی برای جایابی لینک مجازی کمتر می‌گردد.

Joint Service Deployment - Manager Placement (JSD-MP) ۱-۱-۵

برای قرار دادن زنجیره‌ها از همین ایده استفاده می‌کنیم با این تفاوت که تمامی محدودیت‌هایی که در مساله می‌باشند در هر گام برای انتخاب مجموعه‌ای امکان‌پذیر استفاده می‌شوند. در ادامه یک گام به این الگوریتم اضافه می‌کنیم که در آن برای هر زنجیره بعد از قرار گرفتن یک VNFM تخصیص می‌دهیم. برای اینکار مجموعه‌ای امکان‌پذیر از نودهای فیزیکی را انتخاب می‌کنیم و سعی می‌کنیم از بین آن‌ها انتخاب کنیم. در روند این انتخاب از اصول زیر پیروی می‌کنیم:

- اولویت با نود فیزیکی است که روی آن VNFM با ظرفیت خالی وجود دارد.
- از بین نودهایی که ظرفیت خالی دارند اولویت با نودی است که منابع پردازشی بیشتری دارد.

شبه کد

شبه کد سطح بالای الگوریتم پیشنهادی به شرح زیر می‌باشد:

```
for i := 0; i < len(chains); i++ {
    chain := chains[i]
    var feasibleSet []Node
    for i := 0; i < len(chain); i++ {
        if i == 0 {
            feasibleSet = feasibleSetForInstance(i, nil)
        } else {
            selectedNode := nodesWithMaximumReachableNodes(
                feasibleSet(i - 1))[0]
            if selectedNode == nil {
                break
            }
            feasibleSetForInstance(i, selectedNode)
        }
    }
    feasibleManagers := feasibleManagerforChain(chain)
    selectedManager = vnfmWithMoreAvailableResources(
        vnfmWithEmptyCapacity(
```

```

feasibleManagers))[0]
if selectedManager == nil {
    break
}
}
}

```

در اولین گام زنجیره‌ها به ترتیب پیمایش شده و سعی در جایگذاری آن‌ها می‌شود. برای جایگذاری هر زنجیره تمامی نمونه‌های آن به ترتیب پیمایش می‌شوند. برای هر نمونه ابتدا از مجموعه امکان‌پذیر نمونه‌ی قبلی یک نود فیزیکی انتخاب می‌شود که این انتخاب بر اساس تعداد نودهای قابل دسترس از هر نود صورت می‌پذیرد و در ادامه یک مجموعه امکان‌پذیر از نودهای فیزیکی انتخاب می‌شود که این انتخاب بر اساس گام قبلی صورت می‌گیرد. در نظر داشته باشید که انتخاب نودی که نودهای بیشتری از آن قابل دسترس می‌باشند احتمال خالی شدن مجموعه امکان‌پذیر در گام‌های بعدی را کاهش می‌دهد.

در نهایت بعد از اینکه تمامی نمونه‌های زنجیره، جایگذاری شدند VNFN بر اساس معیارهایی که پیشتر شرح داده شد انتخاب می‌شود.

پیچیدگی

جایگذاری هر زنجیره به تعداد نمونه‌های آن زنجیره الگوریتم BFS را اجرا می‌کند. تعداد نمونه‌های هر زنجیره عددی مشخص و کوچک است بنابراین می‌توان از آن به عنوان یک ثابت صرف نظر کرد اما زمان اجرای الگوریتم BFS به اندازه‌ی یال‌های شبکه زیرساخت می‌باشد پس در بدترین حالت (زمانی که مجموعه‌ی امکان‌پذیر همه‌ی نودهای شبکه‌ی زیرساخت را در بر دارد) زمان اجرا از $O(VE)$ می‌باشد و در نهایت الگوریتم از زمان اجرای زیر پیروی می‌کند:

$$O(nVE)$$

enhanced JSD-MP (eJSD-MP) ۲-۱-۵

از آنجایی که مساله‌ی طرح شده به صورت برون خط می‌باشد می‌توان با بررسی ورودی‌های الگوریتم کارایی آن را بهبود داد. برای این منظور زنجیره‌های ورودی را برحسب اندازه‌ی آن‌ها مرتب می‌کنیم. در این مرتب‌سازی تلاش می‌شود که زنجیره‌های بزرگتر که سود

بیشتری دارند زودتر جایابی شوند. به این ترتیب برای زنجیره‌هایی که سود بیشتری دارند منابع بیشتری در اختیار الگوریتم قرار دارد.

نکته‌ای که نیاز به توجه دارد زمان حل است. در دنیای واقعی تغییرات زنجیره‌ها در دیتاسنتر نیاز به اعمال در بازه‌های زمانی کوتاهی دارد که حتی ممکن است زمان ۱۵ دقیقه هم برای آن زیاد باشد. به همین منظور در ادامه بهبودی برای زمان اجرا ارائه می‌کنیم.

الگوریتم JSD-MP در هر گام نود فیزیکی گام قبلی را نهایی کرده و مجموعه امکان‌پذیر این گام را مشخص می‌کند. این امر نیاز به اجرای چند مرحله الگوریتم مسیریابی اول سطح یا BFS دارد که می‌بایست بر روی کل توپولوژی اجرا شود. برای توپولوژی‌هایی که تعداد نودهای آن‌ها زیاد است زمان زیادی برای این عملیات لازم است. برای کاهش این زمان می‌توان درصد مشخصی از زنجیره‌ها را با روش first-fit جایگذاری کرده و زمان اجرای الگوریتم را کاهش داد.

شبه کد

شبه کد سطح بالای الگوریتم پیشنهادی به شرح زیر می‌باشد:

```
sorts.Slice(chains, func (i, j int) {
    return len(chains[i]) > len(chains[j])
})

for i := 0; i < len(chains) * 20 / 100; i++ {
    firstFitPlacement(chain)
}

JSD-MP(chains[len(chains) * 20 / 100:])
```

در اولین گام زنجیره‌ها بر اساس قیمت‌شان مرتب می‌شوند. در نظر داشته باشید که قیمت هر زنجیره ارتباط مستقیمی با تعداد نمونه‌های آن دارد بنابراین با این روش در ابتدا که شبکه زیرساخت خالی است زنجیره‌هایی با ابعاد بزرگتر و سود بیشتر جایگذاری می‌شوند. در ادامه ۲۰ درصد زنجیره‌ها با استفاده از الگوریتم first-fit جایگذاری می‌گردند و برای بقیه زنجیره‌ها از الگوریتم JSD-MP استفاده می‌شود.

پیچیدگی

الگوریتم پیشنهادی برای n زنجیره با یک مرحله مرتب‌سازی با زمان $O(n \log n)$ آغاز می‌شود. برای جایگذاری به روش first-fit می‌توان در نظر گرفت که زمان الگوریتم خطی می‌باشد و قابل صرف نظر کردن است. بنابراین پیچیدگی الگوریتم عبارت است از:

$$O(0.8nVE) + O(n \log n)$$

۲-۵ جمع‌بندی

در این فصل الگوریتم پیشنهادی برای جایگذاری زنجیره‌ها و منابع مدیریتی آن‌ها پیشنهاد شد. این الگوریتم از ایده‌ی الگوریتم [۴] پایه استفاده می‌کند و با مرتب‌سازی اولیه زنجیره‌ها و جایگذاری first-fit تعدادی از زنجیره‌ها کارآیی و زمان اجرا الگوریتم را بهبود می‌دهد. برای جایگذاری منابع مدیریتی سعی می‌شود با استفاده از گواهی‌های مشترک برای زنجیره‌ها هزینه‌ی گواهی را که تاثیر مستقیمی بر تابع هدف دارد کاهش داد. به صورت کلی الگوریتم eJSD-MP بهبودهای زیر را نسبت به الگوریتم JSD-MP ارائه می‌کند:

- با مرتب‌سازی ورودی بر اساس قیمت زنجیره‌ها که ارتباط مستقیمی با تعداد نمونه‌های آن‌ها دارد سعی می‌کند سود بیشتری را از نگاشت نمونه‌ها حاصل کند.
- جایگذاری نسبت مشخصی از زنجیره‌ها با الگوریتم first-fit که باعث صرفه‌جویی در زمان اجرای الگوریتم می‌شود. در نظر داشته باشید که مساله فرض می‌کند شبکه زیرساخت در ابتدا خالی است بنابراین برای جایگذاری زنجیره‌های اولیه نیاز به اخذ تصمیمات پیچیده نداریم.

فصل ششم

ارزیابی

در این مساله هدف بیشینه کردن سود حاصل از پذیرش تقاضاهای زنجیره‌ی کارکرد سرویس می‌باشد که به این ترتیب معیار مقایسه نیز همین پارامتر خواهد بود. باید در نظر داشت که نیازمندی‌های مدیریتی که در این پژوهش مدنظر است در سایر پژوهش‌ها مدنظر نبوده است. راه حل eJSD-MP بهینه نبوده و به همین علت کارایی آن در سناریوهایی با حل بهینه مقایسه می‌شود. سایر پارامترهایی چون تعداد زنجیره‌های پذیرفته شده و ... نیز در این پژوهش ارزیابی می‌گردند.

۱-۶ مقدمه

همانطور که پیشتر بیان شد مساله‌ی اصلی راه حل چند جمله‌ای ندارد. این مساله با استفاده از چهارچوب CPLEX و با زبان جاوا توسعه یافته است. چهارچوب CLPEX توسط شرکت IBM توسعه پیدا کرده است و برای حل مسائل خطی استفاده می‌گردد. این چهارچوب به صورت کلی برای حل مسائل ILP از روش B&C استفاده می‌کند. پیاده‌سازی فرمول‌بندی این مساله در این چهارچوب در پیوست آمده است.

در نهایت قصد داریم در این فصل نشان دهیم الگوریتم eJSD-MP در قیاس با الگوریتم JSD-MP نتایج بهتری را در زمان کمتری ارائه می‌کند و این نتایج برای هر دو الگوریتم به نتایج الگوریتم بهینه نزدیک است.

۲-۶ محیط ارزیابی

به صورت کلی در تمامی ارزیابی‌های این رساله از دو توپولوژی FatTree و USnet استفاده شده است. توپولوژی FatTree یک توپولوژی سازمان یافته است که در ادامه ساختار آن را می‌بینید.

توپولوژی FatTree با مقدار k یک توپولوژی ۳ لایه (هسته، تجمعی و لبه) می‌باشد که:

- هر غلاف از $(k/2)^2$ سرور و ۲ لایه $k/2$ تایی سوئیچ k پورت تشکیل شده است.

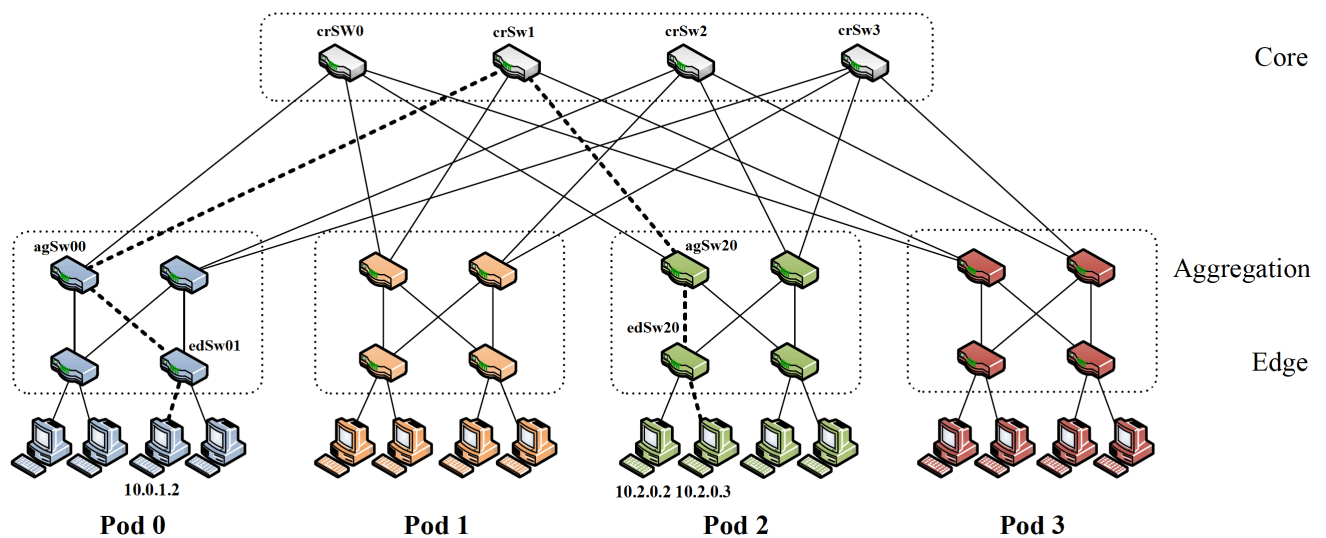
- هر سوئیچ لبه به $k/2$ سرور و $k/2$ سوئیچ تجمعی متصل است.

- هر سوئیچ تجمعی به $k/2$ سوئیچ لبه و $k/2$ سوئیچ هسته متصل است.

- $(k/2)^2$ سوئیچ هسته که هر کدام به k غلاف متصل هستند.

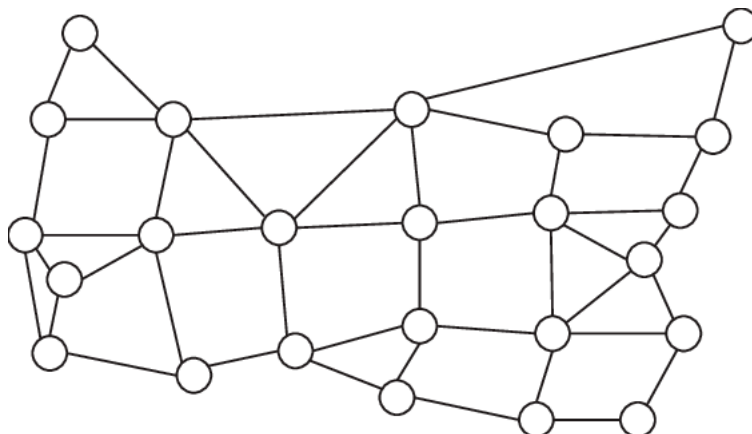
- سوئیچ‌های هسته گره‌های ورودی و خروجی این توپولوژی هستند.

در این توپولوژی فرض می‌کنیم مدیریت هر غلاف تنها می‌تواند در همان غلاف یا غلاف‌های همسایه صورت بگیرد.



شکل ۶-۱: توپولوژی ساختاریافته FatTree

توپولوژی USnet یک تولوپوژی تصادفی می‌باشد که از ۲۴ نود و ۴۳ لینک تشکیل شده است. در پیاده‌سازی فرض شده است که همه‌ی ۲۴ نود سوئیچ هستند و می‌توانند به عنوان گره‌ی ورودی و خروجی اعمال نقش کنند. این توپولوژی دیتاسنتری نبوده و سرورها به صورت تصادفی به سوئیچ‌ها متصل می‌شوند که باعث می‌شود این توپولوژی ماهیت تصادفی داشته باشد.



شکل ۶-۲: توپولوژی تصادفی USnet

۳-۶ معیارهای ارزیابی

همانطور که پیشتر بیان شد معیار اصلی ارزیابی سود حاصل از جایگذاری زنجیره‌ها می‌باشد. پارامترهای زیادی در مساله موثر هستند که در این قسمت به مرور آن‌ها می‌پردازیم.

۱-۳-۶ نسبت سود به هزینه

یکی از ویژگی‌های مهم مساله‌ای طرح شده در نظر گرفتن نیازمندی‌های مدیریتی است. یکی از این نیازمندی‌ها که در تابع هدف هم وجود دارد نیاز به تهیه گواهی برای هر VNFM است. این گواهی هزینه‌ای در بردارد و نیاز است که از آن به درستی استفاده شود و تاجایی که امکان دارد VNFM با ظرفیت خالی نداشت. برای اینکه تخمین درستی از این پارامتر داشته باشیم و بتوانیم از آن در ارزیابی‌های پیش رو استفاده کنیم، موارد زیر را تعریف می‌کنیم:

$$licenseFee/capacity = \text{amortized license cost per instance} \quad (6-1)$$

$$chainPrice/chainLength = \text{amortized price per instance} \quad (6-2)$$

$$\text{amortized price per instance} - \text{amortized license cost per instance} = \text{instance profit} \quad (6-3)$$

در نهایت یکی از پارامترهایی که برای ارزیابی راه‌حل پیشنهادی وجود دارد نسبت سود نمونه به هزینه سرکشن شده گواهی برای هر نمونه می‌باشد. در زمانی که این نسبت عددی کوچک است استفاده نادرست از گواهی‌ها ضرر زیادی می‌زند و شاید بهتر باشد زنجیره‌های کمتری پذیرفته شوند. در حالتی که این نسبت عدد بزرگی باشد می‌توان از این هزینه‌ها صرف‌نظر کرده و تنها منابع مصرفی اجزای مدیریتی مدنظر خواهند بود. در ادامه از این پارامتر تحت عنوان نسبت سود به هزینه یاد می‌کنیم.

۲-۳-۶ سود

سود، اختلاف میان مجموع قیمت زنجیره‌های پذیرفته شده و هزینه‌هایی است که برای گواهی‌ها پرداخت شده است. سود دقیقاً همان تابع هدف مساله است که ارزیابی بر اساس آن صورت می‌گیرد. قیمت زنجیره‌ها پیش از جایگذاری آن‌ها مشخص شده است و فرض می‌کنیم این قیمت با تعداد نمونه‌های داخل زنجیره نسبت مستقیم دارد.

۳-۳-۶ تعداد زنجیره‌های پذیرفته شده

تعداد زنجیره‌هایی است که جایگذاری آن‌ها با موفقیت انجام شده و برای آن‌ها منابع مدیریت نیز تخصیص داده شده است. این معیار در زمانی که پارامتر نسبت سود به هزینه

پایین باشد نمود خوبی از عملکرد الگوریتم نمی‌باشد.

۴-۳-۶ تعداد VNFM‌های استفاده شده

تعداد VNFM‌هایی که برای مدیریت زنجیره‌ها تخصیص داده می‌شوند نمایش دهنده‌ی تعداد گواهی‌های استفاده شده است. این معیار در زمانی که پارامتر نسبت سود به هزینه بالا باشد نمود خوبی از عملکرد الگوریتم نمی‌باشد.

۴-۶ محیط ارزیابی

برای ارزیابی از زنجیره‌های تصادفی استفاده می‌شود و هر نمونه از ارزیابی میانگین ۱۰ اجرا می‌باشد. برای تولید زنجیره‌های تصادفی از ابزاری استفاده می‌شود که برای همین پژوهش توسعه یافته است و زنجیره‌های خطی با طول تصادفی تولید می‌کند. نمونه‌های داخل زنجیره‌ها دارای نوع می‌باشند که به صورت تصادفی از لیست زیر انتخاب می‌شوند:

types:

- name: ingress
 - cores: 0
 - ram: 0
 - ingress: true
 - manageable: false
- name: egress
 - cores: 0
 - ram: 0
 - egress: true
 - manageable: false
- name: vFW
 - cores: 2
 - ram: 2
 - manageable: true
- name: vNAT
 - cores: 2
 - ram: 4
 - manageable: true

```

- name: vIDS
  cores: 2
  ram: 2
  manageable: true
- name: vDPI
  cores: 2
  ram: 4
  manageable: true

```

زنجیره‌های تولید شده دارای گرهی آغازی و پایانی می‌باشند و ترافیک عبوری از آن‌ها ۲۵۰ واحد است. تنظیمات زیر برای VNFM‌ها در نظر گرفته شده است.

```

ram: 4
cores: 2
capacity: 10
radius: 100
bandwidth: 1
licenseFee: 100

```

تنظیمات زیر برای زیرساخت فیزیکی در نظر گرفته شده است:

```

ram: 10 - 48
cores: 100 - 700
bandwidth: 40000

```

تمامی ارزیابی‌ها روی سیستمی با مشخصات زیر انجام شده‌اند:

- AMD Ryzen Threadripper 1950X 16-Core Processor
- 22 GB of RAM
- 100 GB of non-SSD Storage

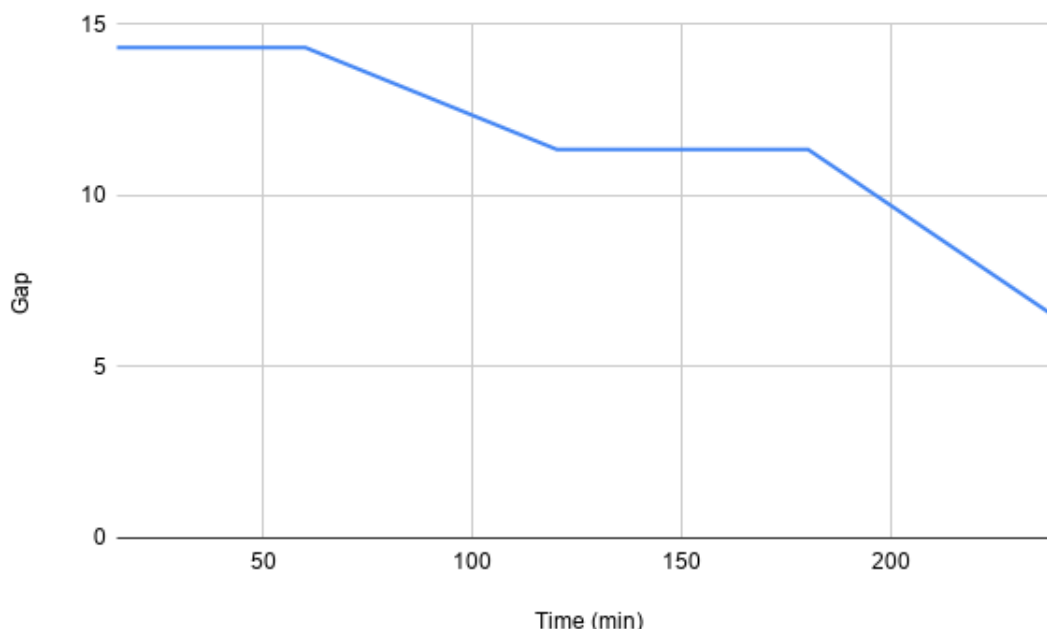
در نهایت برای بازتولید نتایج تمامی کدها و تنظیمات در [۳] موجود است.

۵-۶ نتایج ارزیابی

ابتدا زمان حل مسالهی بهینه و تاثیر نسبت سود به هزینه در مساله را بررسی می‌کنیم. به این ترتیب می‌توانیم شرایط ارزیابی که در آن عمل می‌کنیم را استدلال کنیم. در ادامه به گزارش و تحلیل نتایج می‌پردازیم.

۱-۵-۶ زمان حل بهینه

با استفاده از ۱۳۰ زنجیره‌ی تصادفی که طولی بین ۳ تا ۷ دارند و توپولوژی FatTree با مقدار k برابر ۸ قصد داریم زمان حل راه حل بهینه و شکاف بهینه^۱ آن را ارزیابی کنیم. برای این ارزیابی نسب سود به هزینه برابر ۹ فرض شده است.



شکل ۳-۶: شکاف بهینه الگوریتم بهینه بر اساس زمان اجرا (بر حسب دقیقه)

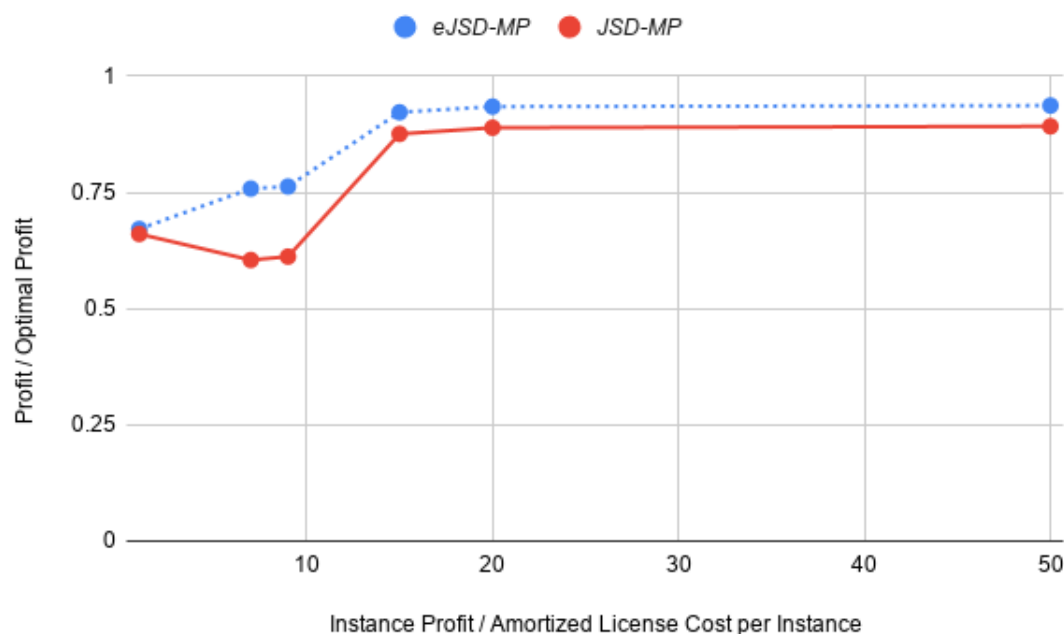
شکاف بهینه برای ۱۰۰ زنجیره در ۱۵ دقیقه با شرایط فوق برابر با ۴ درصد می‌باشد. بنابراین در سایر ارزیابی‌ها الگوریتم بهینه را تا ۱۵ دقیقه محدود کرده و تعداد زنجیره‌ها را به بیش از ۱۰۰ افزایش نمی‌دهیم.

همانطور که در نمودار ۳-۶ مشاهده می‌شود زمان حل مسالهی بهینه برای ۱۳۰ زنجیره نسبت به ۱۰۰ زنجیره جهش بزرگی داشته و بعد از ۴ ساعت به شکاف بهینه زیر ۱۰ درصد می‌رسیم. به این ترتیب استفاده از راه‌حل بهینه ممکن است زمان‌بر باشد و نیاز به پیاده‌سازی یک راه‌حل مکاشفه‌ای است.

^۱ Optimality Gap

۲-۵-۶ نسبت سود به هزینه

در ادامه راه حل eJSD-MP و راه حل JSD-MP را با نسبت‌های مختلف سود به هزینه مورد آزمون قرار می‌دهیم. در این آزمون‌ها از ۱۰۰ زنجیره با طول‌های تصادفی ۳ تا ۷ استفاده می‌کنیم. توپولوژی مورد استفاده FatTree با مقدار k برابر با ۸ می‌باشد. در این آزمایش‌ها نسبت سود حاصل از هر الگوریتم به الگوریتم بهینه سنجیده شده و در نمودار آمده است. در نظر داشته باشید که این نسبت به صورت عددی بین ۰ تا ۱ گزارش شده است. مساله بهینه با زمان ۱۵ دقیقه محدود شده است.



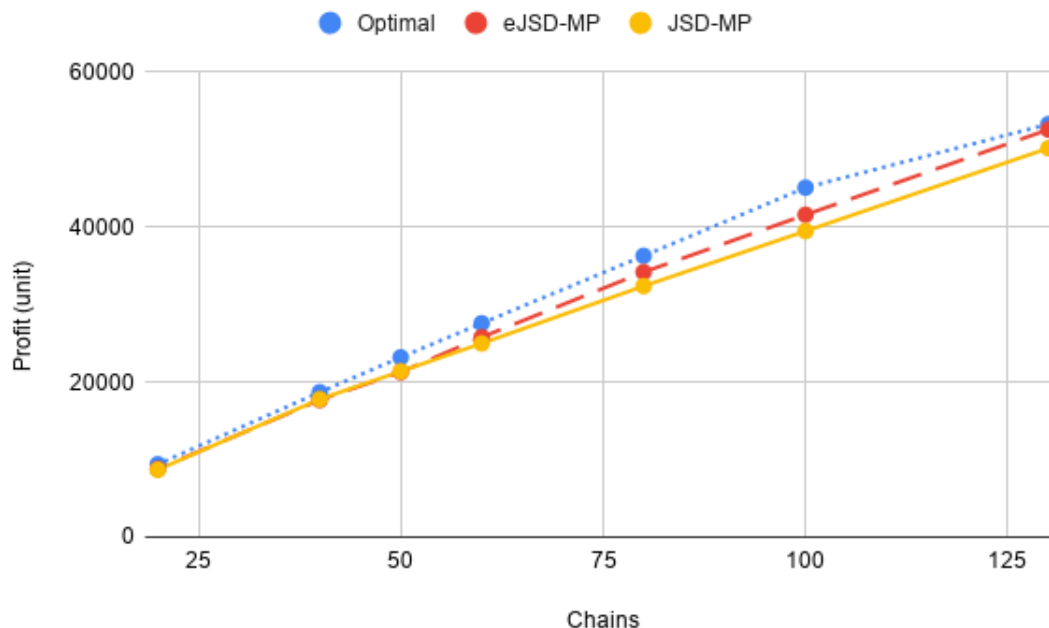
شکل ۴-۶: کارایی الگوریتم‌های eJSD-MP و JSD-MP در نسبت‌های مختلف سود به هزینه

همانطور که در نمودار ۴-۶ دیده می‌شود الگوریتم eJSD-MP بهتر از الگوریتم JSD-MP عمل می‌کند. این امر زمانی که نسبت سود به هزینه بزرگتر است بیشتر دیده می‌شود. در ادامه برای تمامی ارزیابی‌ها از نسبت سود به هزینه ۹ استفاده می‌کنیم که از نظر فنی عدد معقولی بوده و تاثیر در نظر گرفتن هزینه گواهی را از بین نمی‌برد.

۳-۵-۶ زنجیره‌ها در توپولوژی FatTree

در تمامی این ارزیابی‌ها از نسبت سود به هزینه ۹ استفاده کرده و زنجیره‌ها را در توپولوژی FatTree جایگذاری می‌کنیم. در این ارزیابی تعداد زنجیره‌ها را تغییر می‌دهیم اما همواره طول زنجیره‌ها بین ۳ تا ۷ می‌باشد. همانطور که بیان شد الگوریتم بهینه برای تمامی حالت‌ها تا ۱۵ دقیقه محدود شده است. برای حالت ۱۳۰ برای رسیدن به شکاف بهینه

زیر ۱۰ درصد نیاز به ۴ ساعت زمان است.



شکل ۵-۶: سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree

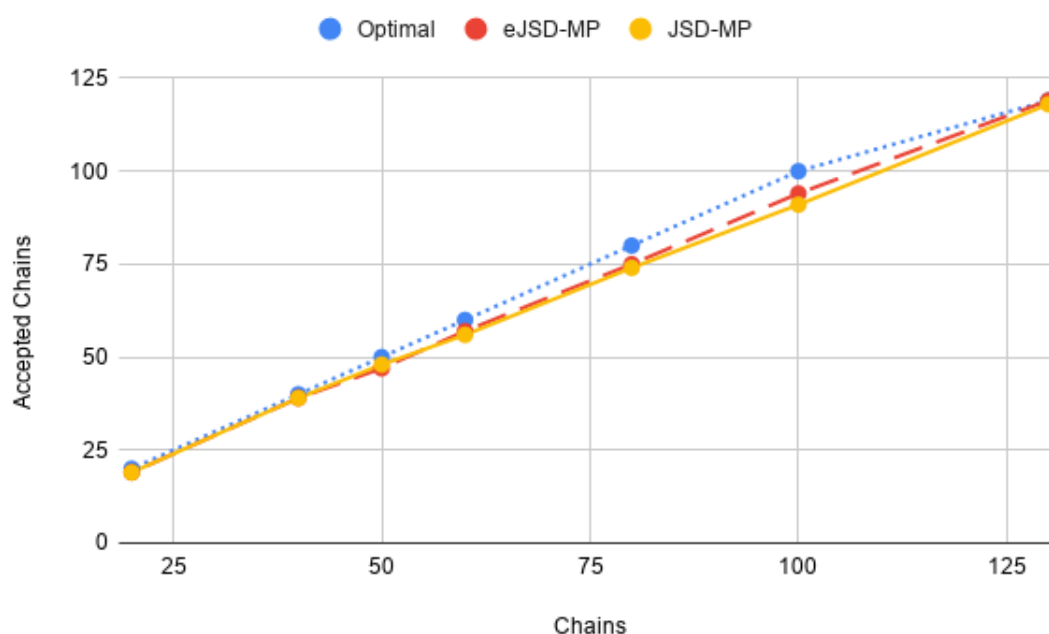
همانطور که در نمودار ۵-۶ مشخص است با افزایش تعداد زنجیره‌ها الگوریتم JSD-MP از الگوریتم eJSD-MP بدتر عمل می‌کند، به این ترتیب که سود حاصل از پذیرفتن زنجیره‌ها در الگوریتم eJSD-MP بیشتر است. یکی از موارد مهم در زمانی که ۱۳۰ زنجیره وجود دارد نزدیکی جواب بهینه به جواب الگوریتم eJSD-MP است. در این حالت به علت پیچیدگی مساله همانطور که صحبت شد تولید جواب با شکاف بهینه مناسب زمان زیادی می‌برد، بنابراین جوابی که در این حالت استفاده شده نسبت به سایر نقاط ۲ درصد شکاف بهینه بیشتری دارد.

الگوریتم eJSD-MP به طور میانگین ۵ درصد از الگوریتم JSD-MP بهتر عمل کرده و سود خالص بیشتری تولید می‌کند. دلیل این بهبود در سود نهایی به خاطر مرتب‌سازی زنجیره‌ها بر اساس قیمت آن‌ها می‌باشد. الگوریتم eJSD-MP پیش از جایگذاری زنجیره‌ها، آن‌ها را بر اساس قیمت‌شان مرتب می‌کند که این امر به آن اجازه می‌دهد در زمانی که هنوز زیرساخت خالی است زنجیره‌هایی که سود بیشتری دارند را جایگذاری نماید. در حالی که الگوریتم JSD-MP زیرساخت را با زنجیره‌هایی که سود زیادی ندارند پر کرده و امکان جایگذاری زنجیره‌های پر سود را از دست می‌دهد. همانطور که پیشتر نیز بیان شده بود، روند دو الگوریتم با افزایش تعداد زنجیره‌ها صعودی بوده و جواب حاصل هر دو از جواب بهینه فاصله زیادی نمی‌گیرد.

نمودارهای ۶-۶ و ۷-۶ بیشتر جنبه‌ی اطلاعی دارند و تعداد زنجیره‌های پذیرفته شده و

جدول ۶-۱: سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree

تعداد زنجیره‌ها	الگوریتم بهینه	سود نهایی	eJSD-MP	نسبت به بهینه	JSD-MP	سود نهایی	نسبت به بهینه
#							
130	53300	52600	98.69%	50200	94.18%		
100	45100	41600	92.24%	39500	87.58%		
80	36300	34200	94.21%	32400	89.26%		
60	27600	25800	93.48%	25000	90.58%		
50	23200	21300	91.81%	21400	92.24%		
40	18700	17700	94.65%	17800	95.19%		
20	9400	8800	93.62%	8700	92.55%		

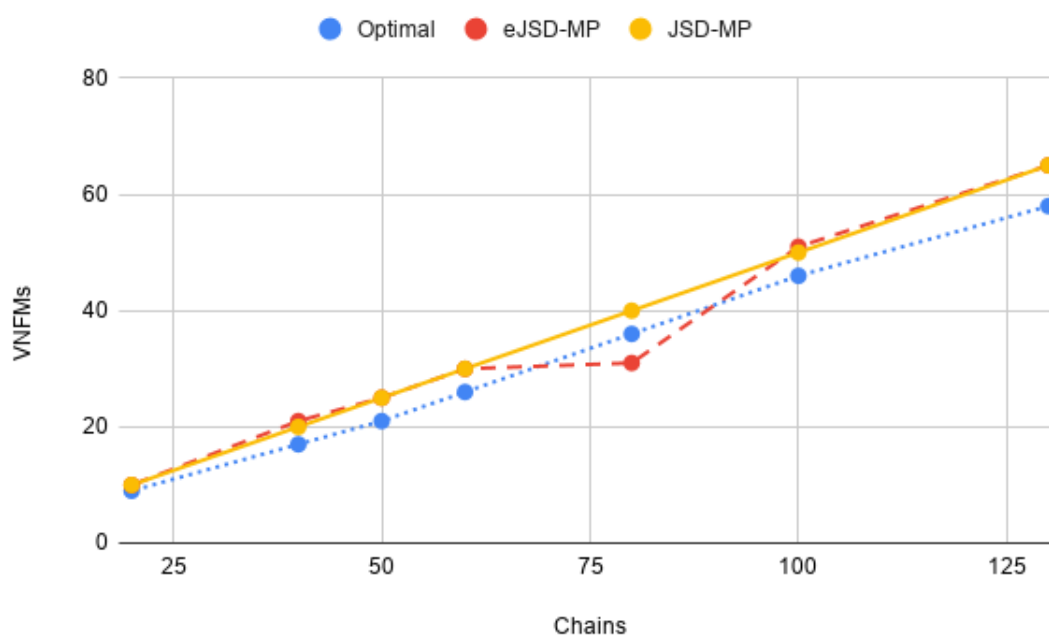


شکل ۶-۶: تعداد زنجیره‌های پذیرفته شده الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree

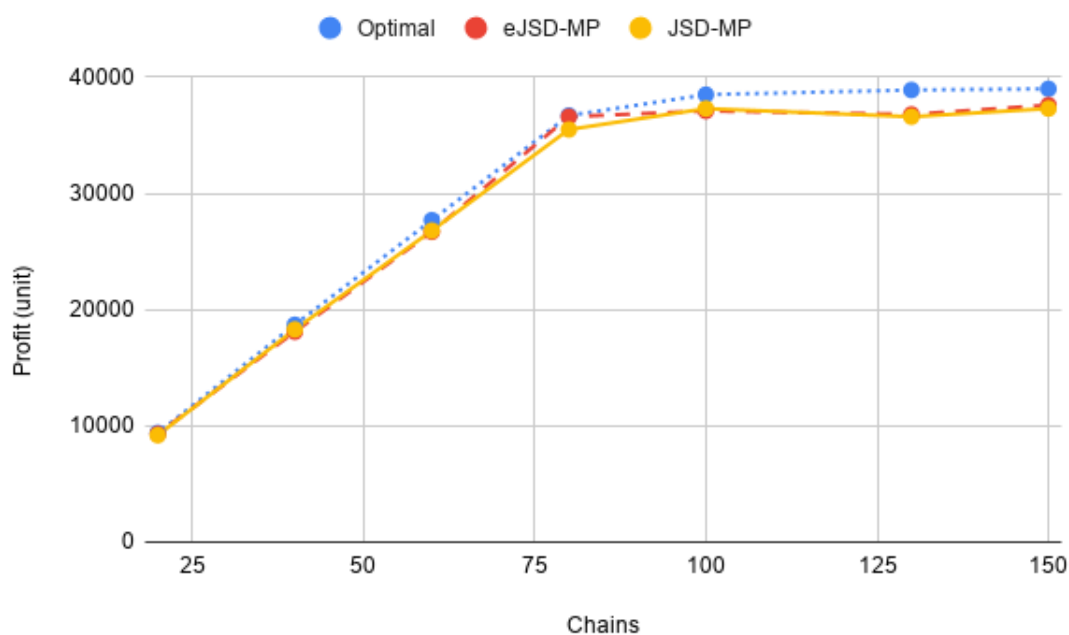
تعداد VNFM‌های استفاده شده را نشان می‌دهند. همانطور که در تعریف مساله نیز گفته شده است، این دو پارامتر در سود نهایی تاثیر دارند اما سود نهایی به ضرایب آن‌ها نیز وابسته است.

۴-۵-۶ زنجیره‌ها در توپولوژی USnet

در تمامی این ارزیابی‌ها از نسبت سود به هزینه ۹ استفاده کرده و زنجیره‌ها را در توپولوژی USnet جایگذاری می‌کنیم. در این ارزیابی‌ها تعداد زنجیره‌ها را تغییر می‌دهیم اما همواره طول زنجیره‌ها بین ۳ تا ۷ می‌باشد.



شکل ۶-۷: تعداد VNFMs الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی FatTree



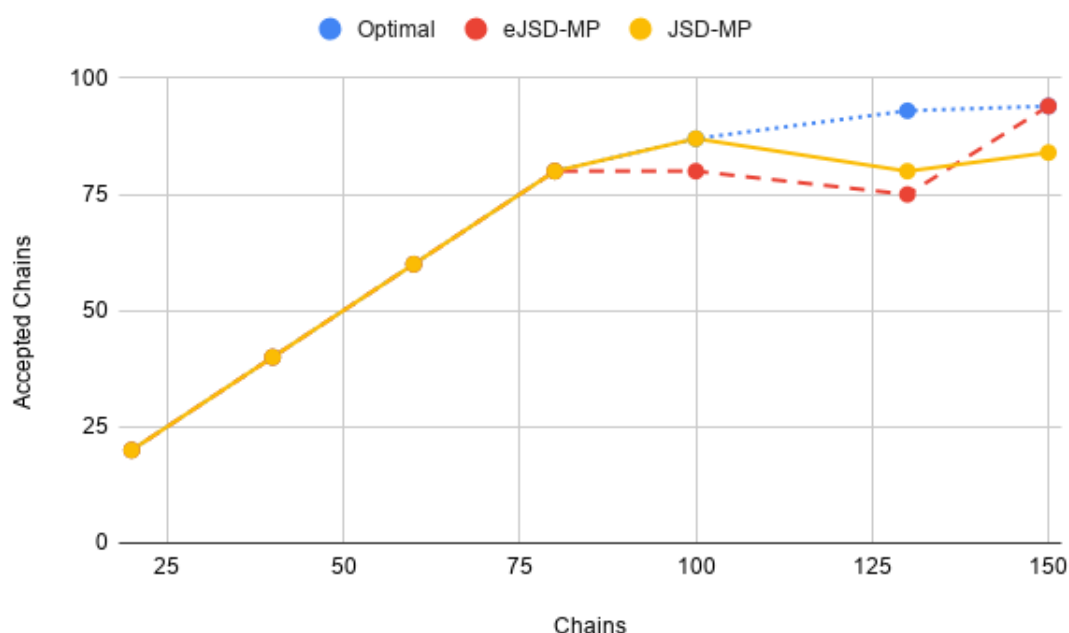
شکل ۶-۸: سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet

از آنجایی که تعداد لینک‌ها و نودهای این توپولوژی کم می‌باشد الگوریتم‌های eJSD-MP و JSD-MP هر دو مشابه یکدیگر عمل می‌کنند که این امر در نمودار ۶-۸ نیز قابل رویت است. اگر بخواهیم این امر را دقیق‌تر بررسی کنیم در این توپولوژی ظرفیت کم بوده و

نمی‌توان تعداد زیادی از زنجیره‌های پرسود را جایگذاری کرد این در حالی است که این توپولوژی محدودیت خاصی برای جایگذاری منابع مدیریتی ندارد پس می‌توان به سادگی منابع مدیریتی را جایگذاری کرد. با این تفاسیر مرتب‌سازی اولیه باعث می‌شود که به علت نبود منابع کافی تعداد کمی زنجیره‌ی پرسود جایگذاری شوند. این در حالی است که الگوریتم JSD-MP تعداد زنجیره‌های کوچک بیشتری را جایگذاری می‌کند و منابع مدیریتی را نیز به سادگی به آن‌ها تخصیص می‌دهد.

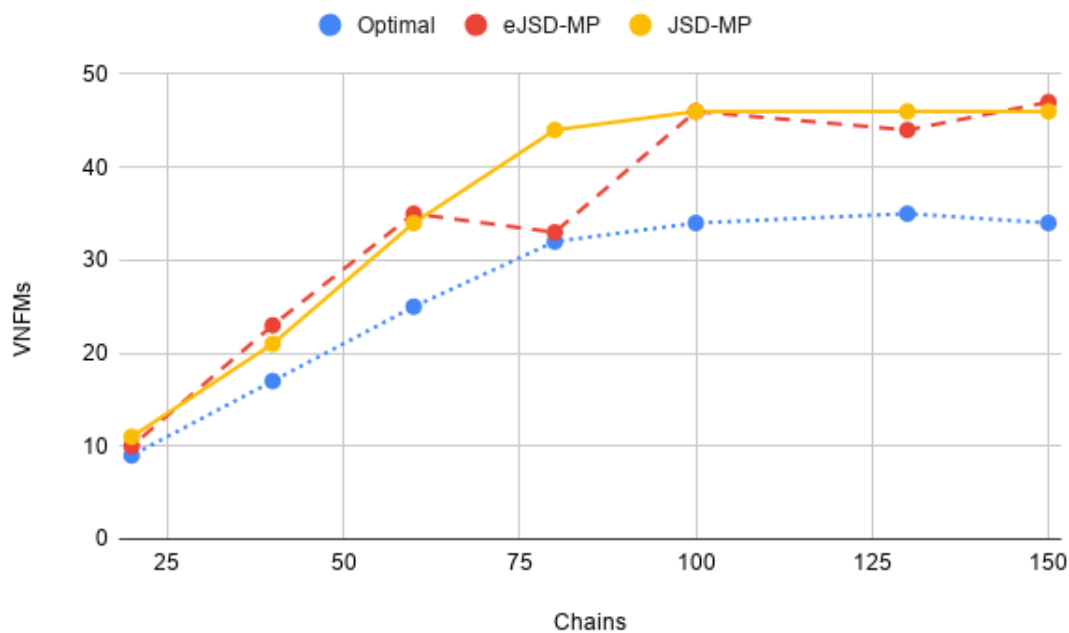
جدول ۶-۲: سود نهایی الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet

تعداد زنجیره‌ها	الگوریتم بهینه	eJSD-MP	JSD-MP		
#	سود نهایی	سود نهایی	نسبت به بهینه	سود نهایی	نسبت به بهینه
150	39000	37600	96.41%	37300	95.64%
130	38900	36800	94.6%	36600	94.09%
100	38500	37100	96.36%	37300	96.88%
80	36700	36600	99.73%	35500	96.73%
60	27700	26700	96.39%	26800	96.75%
40	18700	18100	96.79%	18300	97.86%
20	9400	9300	98.94%	9200	97.87%



شکل ۶-۹: تعداد زنجیره‌های پذیرفته شده الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet

نمودارهای ۶-۱۰ و ۶-۹ بیشتر جنبه‌ی اطلاعی دارند و تعداد زنجیره‌های پذیرفته شده



شکل ۶-۱۰: تعداد VNFMs الگوریتم‌های بهینه، eJSD-MP و JSD-MP برای توپولوژی USnet

و تعداد VNFMs استفاده شده را نشان می‌دهند. همانطور که در تعریف مساله نیز گفته شده است، این دو پارامتر در سود نهایی تاثیر دارند اما سود نهایی به ضرایب آن‌ها نیز وابسته است.

۶-۶ ارزیابی زمان اجرا

همانطور که پیشتر بیان شده بود یکی از بهبودهای الگوریتم eJSD-MP نسبت به الگوریتم JSD-MP زمان اجرای آن می‌باشد. در این قسمت قصد داریم این موضوع را مورد آزمایش قرار دهیم. به این منظور از توپولوژی FatTree با مقدار k برابر با ۱۰ استفاده می‌کنیم. با توجه به ابعاد این توپولوژی امکان حل آن با الگوریتم بهینه بر روی سیستم eJSD-MP وجود ندارد.

در این ارزیابی از ۱۰۰ زنجیره‌ی ورودی استفاده می‌شود. الگوریتم eJSD-MP در زمان ۵ دقیقه ۱۰۰ زنجیره را جایگذاری کرده و از ۳۳ نمونه VNFMs استفاده می‌کند. در حالی که الگوریتم JSD-MP در زمان ۷ دقیقه تنها ۹۹ زنجیره را جایگذاری کرده و از ۵۳ نسخه VNFMs استفاده می‌کند.

همانطور که بیان شد، با توجه به این که در ابتدای امر تمامی زیرساخت خالی می‌باشد می‌توان از یک الگوریتم ساده برای جایگذاری استفاده کرد و به این ترتیب بدون تاثیر در جواب نهایی زمان حل مساله را کاهش داد.

در این حالت ۲۰ زنجیره‌ی ابتدایی به صورت first-fit جایگذاری شدند.

۷-۶ جمع‌بندی

در این فصل راه‌حل eJSD-MP را ارزیابی کرده و نشان دادیم که این راه‌حل به طور میانگین برای توپولوژی‌های FatTree و USnet ۹۰ درصد جواب بهینه را ارائه می‌دهد. در ادامه نشان داده شد این الگوریتم از الگوریتم JSD-MP در توپولوژی FatTree به صورت میانگین ۵ درصد بهتر عمل می‌کند. و در توپولوژی USnet به طور میانگین ۱ درصد بهبود عملکرد دارد.

در ادامه در ارزیابی‌ها نشان داده شد که الگوریتم پیشنهادی برای نگاشت منابع مدیریتی به صورت میانگین ۷۰ درصد الگوریتم بهینه عملکرد دارد که برای نسبت سود به هزینه‌ای که در عمل وجود دارد عددی مناسب است.

در نظر داشته باشید که هر دو الگوریتم پیشنهادی جواب‌هایی نزدیک به جواب بهینه تولید می‌کنند به این معنا که به صورت میانگین جواب‌های تولید شده حداکثر ۱۰ درصد با جواب بهینه در سود نهایی اختلاف دارند.

فصل هفتم

نتیجه‌گیری و کارهای آینده

در این فصل نتیجه‌گیری و جمع‌بندی پژوهش‌ها و فعالیت‌های انجام شده در این رساله آورده شده است. همچنین با توجه به کارهای انجام شده، کارهای آینده برای پیشبرد این پژوهش برشمرده شده‌اند.

۱-۷ جمع‌بندی و نتیجه‌گیری

همانطور که پیشتر بیان شد پژوهش‌های زیادی به مسائل جایگذاری زنجیره‌های کارکرد سرویس پرداخته‌اند اما هیچ یک از آن‌ها نیازمندی مدیریتی برای این زنجیره‌ها همانطور که در [۱] بیان شده است را در نظر نگرفته‌اند.

از آنجایی که مانیتورینگ و مدیریت زنجیره‌های در مراکز داده اهمیت زیادی دارد، نیاز است تا در هنگام جایگذاری زنجیره‌ها منابع مدیریتی آن‌ها را نیز مدنظر قرار داد. در این پژوهش تلاش شد مساله‌ای جامع و نزدیک به واقعیت برای جایگذاری توامان زنجیره‌ها و منابع مدیریتی آن‌ها طرح و روشی برای حل آن ارائه شود.

پژوهش‌های حاضر منابع مدیریتی را لحاظ نکرده‌اند و این امر تنها در [۲] مدنظر قرار داده شده است. این در حالی است که این پژوهش نیز منابع مدیریتی را به صورت توامان با زنجیره‌ها جایگذاری نکرده است و منابع مدیریتی را برای یک شبکه‌ی در حال کار نگاشت کرده است.

در این پژوهش، در ابتدا یک مدل بهینه‌سازی MILP برای مساله‌ی طرح شده ارائه شد که هدف آن بیشینه کردن میزان سود حاصل از جایگذاری زنجیره‌ها و تخصیص منابع مدیریتی به آن‌ها بود. در این مدل سعی شد تا تمامی سیاست‌های یک مرکز داده‌ای در جایگذاری کارکردهای مجازی و منابع مدیریتی آن‌ها مدنظر باشد. با توجه به اینکه مساله‌ی طرح شده یک مساله‌ی NP-Hard است و پیدا کردن راه‌حل بهینه از نظر محاسباتی و زمانی مقدور نیست. در این پژوهش یک الگوریتم اکتشافی ارائه شد که دارای دو بخش اصلی نگاشت زنجیره و نگاشت منابع مدیریتی است.

در بخش نگاشت زنجیره‌ها ابتدا زنجیره‌ها بر اساس سودشان مرتب می‌شوند در ادامه بر اساس الگوریتم [۴] جایگذاری شده و در نهایت بعد از نگاشت زنجیره VNFM آن انتخاب می‌گردد. در این انتخاب سعی می‌شود از VNFM با ظرفیت خالی و منابع آزاد استفاده شود.

در نهایت در قسمت ارزیابی این الگوریتم با حالت بهینه و الگوریتم [۴] مقایسه می‌شود که نتیجه نشان می‌دهد سود حاصل از جایگذاری با استفاده از این الگوریتم به طور میانگین ۹۰ درصد الگوریتم بهینه بوده و ۵ درصد نسبت به [۴] افزایش دارد. در نظر داشته باشید که این الگوریتم منابع مدیریتی را نیز مدنظر قرار می‌دهد که پیشتر در پژوهش‌ها اشاره‌ای

به آن‌ها نشده است. این الگوریتم با بیشتر شدن تعداد زنجیره‌ها کارایی بیشتری از خود نشان می‌دهد چرا که قسمت مرتب‌سازی با افزایش تعداد زنجیره‌ها باعث می‌گردد زنجیره‌هایی با سود بیشتر جایگذاری شوند و به سود کلی عملیات کمک می‌کند. مورد مهم دیگر در ارزیابی چگونگی مصرف منابع مدیرتی است که به صورت میانگین الگوریتم پیشنهادی برای این نگاشت ۷۰ درصد الگوریتم بهینه عملکرد دارد.

۱-۱-۷ کارهای آینده

در این پژوهش تلاش شد تا ابعاد مختلف جاسازی زنجیره‌ها در نظر گرفته شود. با این وجود مواردی وجود دارد که در نظر گرفتن آن‌ها می‌تواند باعث دقیق‌تر شدن مساله و نزدیک شدن آن به واقعیت شود. در ادامه این موارد مرور می‌شوند:

- در نظر گرفتن نیاز به VNFO برای مدیریت تعاملات زنجیره‌ها
- در نظر گرفتن نیاز ارتباط NVFI-PoP با VIM
- به اشتراک‌گذاری کارکردهای مجازی و در نظر گرفتن هزینه‌ی گواهی برای آن‌ها
- در نظر گرفتن EMها
- بهبود الگوریتم نگاشت منابع مدیریتی

منابع و مراجع

- [1] Etsi gs nfv-man 001 v1.1.1: Network function virtualization (nfv): Management and orchestration. Technical report, december 2014.
- [2] Abu-Lebdeh, Mohammad, Naboulsi, Diala, Glitho, Roch, and Tchouati, Constant Wette. On the placement of VNF managers in large-scale and distributed NFV systems. *IEEE Transactions on Network and Service Management*, 14(4):875–889, dec 2017.
- [3] Alvani, Parham. The road to master of science degree, 2019.
- [4] Bari, Md. Faizul, Chowdhury, Shihabur Rahman, Ahmed, Reaz, and Boutaba, Raouf. On orchestrating virtual network functions. In *2015 11th International Conference on Network and Service Management (CNSM)*. IEEE, November 2015.
- [5] Chen, Yu-Jia, Wang, Li-Chun, Lin, Feng-Yi, and Lin, Bao-Shuh Paul. Deterministic quality of service guarantee for dynamic service chaining in software defined networking. *IEEE Transactions on Network and Service Management*, 14(4):991–1002, dec 2017.
- [6] Eramo, V., Tosti, A., and Miucci, E. Server resource dimensioning and routing of service function chain in NFV network architectures. *Journal of Electrical and Computer Engineering*, 2016:1–12, 2016.
- [7] Eramo, Vincenzo, Miucci, Emanuele, Ammar, Mostafa, and Lavacca, Francesco Giacinto. An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures. *IEEE/ACM Transactions on Networking*, 25(4):2008–2025, aug 2017.

- [8] Ghaznavi, Milad, Shahriar, Nashid, Kamali, Shahin, Ahmed, Reaz, and Boutaba, Raouf. Distributed service function chaining. *IEEE Journal on Selected Areas in Communications*, 35(11):2479–2489, nov 2017.
- [9] Huang, Huawei, Li, Peng, Guo, Song, Liang, Weifa, and Wang, Kun. Near-optimal deployment of service chains by exploiting correlations between network functions. *IEEE Transactions on Cloud Computing*, pages 1–1, 2017.
- [10] Jang, Insun, Suh, Dongeun, Pack, Sangheon, and Dan, Gyorgy. Joint optimization of service function placement and flow distribution for service function chaining. *IEEE Journal on Selected Areas in Communications*, 35(11):2532–2541, nov 2017.
- [11] Ma, Wenrui, Beltran, Jonathan, Pan, Zhenglin, Pan, Deng, and Pissinou, Niki. SDN-based traffic aware placement of NFV middleboxes. *IEEE Transactions on Network and Service Management*, 14(3):528–542, sep 2017.
- [12] Mijumbi, Rashid, Serrat, Joan, Gorricho, Juan-Luis, Bouten, Niels, Turck, Filip De, and Boutaba, Raouf. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials*, 18(1):236–262, 2016.
- [13] Pham, Tuan-Minh, Nguyen, Thi-Thuy-Lien, Fdida, Serge, and Binh, Huynh Thi Thanh. Online load balancing for network functions virtualization. In *2017 IEEE International Conference on Communications (ICC)*. IEEE, may 2017.
- [14] Yu, Ruozhou, Xue, Guoliang, and Zhang, Xiang. QoS-aware and reliable traffic steering for service function chaining in mobile networks. *IEEE Journal on Selected Areas in Communications*, 35(11):2522–2531, nov 2017.

واژه‌نامه

Interface	رابط
Application	کاربرد
Optimality Gap	شکاف بهینه
Edge	یال
Node	گره
Virtualization	مجازی‌سازی
Constraint	محدودیت
Heuristic	مکاشفه‌ای
Offline	برون خط
Online	بر خط
Image	تصویر
Classifier	دسته‌بند
Forwarding	جلورانی
Function	کارکرد
Pod	غلاف
Aggregation	تجمعی

Edge	لبه
Instance	نمونه
Feasible Set	مجموعه امکان پذیر
Monitoring	نظارت
Data Center	مرکز داده

Abstract

In the old times, Network providers use hardware network functions to create their service chains, but a change in this manner is difficult and may cause many service distribution. SFC and NFV is the solution to this difficulty. By using SFC and NFV, providers can provision chains dynamically and then change them in runtime. One of the main requirements is management and monitoring for the chains. In this research, we consider the chain acceptance problem subject to management resources. In the first step, we formulate problem with ILP and then implement it in CPLEX framework. As we know, ILP problems are NP-Hard, so we need an NP solution to the problem. In this research, we create a heuristic algorithm and compare its result with the optimal solution. In the end, the heuristic solution produces near-optimal results in the polynomial time.

Key Words:

NFV, SFC, Optimization, ILP



**Amirkabir University of Technology
(Tehran Polytechnic)**

Department of Computer Engineering

MSc Thesis

**Virtualized Network Service Function
Chaining Subject to Management
Resource Constraint**

By

Parham Alvani

Supervisor

Prof. Bahador Bakhshi

September 2019