

دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر
گروه مهندسی برق



هماهنگ سازی مسیریابی و جایابی توابع مجازی شبکه در شبکه‌های نسل پنجم

پایان نامه برای دریافت درجه کارشناسی ارشد در رشته مهندسی مخابرات
گرایش شبکه‌های مخابراتی

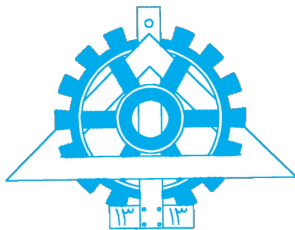
گلاره حاصل مهری

استاد راهنما

دکتر وحید شاه منصوری

شهریور ۱۳۹۸





دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر
گروه مهندسی برق



هماهنگ سازی مسیریابی و جایابی توابع مجازی شبکه در شبکه‌های نسل پنجم

پایان نامه برای دریافت درجه کارشناسی ارشد در رشته مهندسی مخابرات
گرایش شبکه‌های مخابراتی

گلاره حاصل مهری

استاد راهنما

دکتر وحید شاه منصوری

شهریور ۱۳۹۸



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر
گروه مهندسی برق



گواهی دفاع از پایان‌نامه کارشناسی ارشد

هیأت داوران پایان‌نامه کارشناسی ارشد آقای / خانم گلاره حاصل مهری به شماره دانشجویی ۸۱۰۱۹۵۰۶۷ در رشته مهندسی مخابرات - گرایش شبکه‌های مخابراتی را در تاریخ با عنوان « هماهنگ سازی مسیریابی و جایابی توابع مجازی شبکه در شبکه‌های نسل پنجم »

به عدد	به حروف
<input type="text"/>	<input type="text"/>

با نمره نهایی

ارزیابی کرد.

و درجه

ردیف	مشخصات داوران	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضاء
۱	استاد راهنما	دکتر وحید شاه منصوری	استادیار	دانشگاه تهران	
۲	استاد داور داخلی	دکتر علی اعظم عباسفر	دانشیار	دانشگاه تهران	
۳	استاد مدعو	دکتر سید پویا شریعت پناهی	استادیار	پژوهشگاه دانش های بنیادی	

نام و نام خانوادگی معاون تحصیلات تکمیلی و نام و نام خانوادگی معاون آموزشی و تحصیلات

تکمیلی پردیس دانشکده‌های فنی:

پژوهشی دانشکده / گروه:

تاریخ و امضاء:

تاریخ و امضاء:

تعهدنامه اصالت اثر

باسمه تعالی

اینجانب گلاره حاصل مهری تأیید می‌کنم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب است و به دستاوردهای پژوهشی دیگران که در این نوشته از آنها استفاده شده است مطابق مقررات ارجاع گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نشده است.

نام و نام خانوادگی دانشجو: گلاره حاصل مهری

تاریخ و امضای دانشجو:

کلیه حقوق مادی و معنوی این اثر
متعلق به دانشگاه تهران می باشد.

قدردانی

از زحمات بی دریغ استاد راهنمای خود، جناب آقای دکتر وحید شاه منصوری، صمیمانه تشکر و قدردانی می‌کنم که در طول انجام این پایان‌نامه با نهایت صبوری همواره راهنما و مشوق من بودند و قطعاً بدون راهنمایی‌های ارزنده ایشان، این مجموعه به انجام نمی‌رسید.

گلاره حاصل‌مهری

شهریور ۱۳۹۸

چکیده

نسل پنجم^۱ شبکه‌های بی‌سیم، به عنوان بستری برای افزایش قابل توجه ظرفیت شبکه، اتصال گسترده دستگاه‌ها توأم با تأخیر و هزینه کم و همچنین صرفه‌جویی قابل توجه در مصرف انرژی مطرح شده است. در این راستا طراحان، مدیریت و ارتقا پیوسته شبکه نیازمند بهره‌گیری از فناوری‌های نوین است. ارائه‌دهندگان خدمات به طور گسترده از فناوری‌هایی مانند مجازی‌سازی توابع شبکه^۲ و شبکه نرم‌افزار محور^۳، استفاده می‌کنند تا از مزایای کاهش هزینه‌های اولیه و هزینه‌های عملیاتی بهره‌مند شوند.

در این پژوهش، مساله جایابی^۴ توابع مجازی شبکه^۵ بر روی منابع موجود و همچنین مسیریابی^۶ داده از میان این توابع جهت اجرای زنجیره سرویسی خاص، مورد بررسی قرار می‌گیرد. در این راستا، شبکه‌ای متشکل از سوئیچ‌ها و سرورها با مشخصات معین در نظر گرفته شده و مساله بهینه‌سازی با هدف حداقل کردن مجموع هزینه‌های استفاده از لینک‌ها و سرورها و با در نظر گرفتن قیود مختلف تشکیل شده است. این قیود شامل قیود شبکه‌ای، قیود ساختاری و قید حداکثر ازدحام قابل تحمل روی هر لینک می‌باشند. جهت برآورد کردن هزینه‌ها نیز، هم دیدگاه فراهم‌آورنده سرویس و هم دیدگاه سرمایه‌گذار و صاحب زیرساخت در نظر گرفته شده است. در ادامه دو الگوریتم اکتشافی برای حل مساله بهینه‌سازی مذکور ارائه شده اند و یکی از آن‌ها به کمک نرم‌افزار شبیه سازی شده و عملکرد آن با حل در حالت بهینه مقایسه شده است.

واژگان کلیدی: نسل پنجم مخابرات سلولی، شبکه نرم‌افزار محور، مجازی‌سازی توابع شبکه، مسیریابی، الگوریتم اکتشافی

¹5th Generation

²Network Function Virtualization

³Software-Defined Networking

⁴placement

⁵Virtualized Network Function

⁶routing

فهرست مطالب

ت	فهرست تصاویر
۱	فصل ۱: مقدمه
۱-۱	مقدمه
۱-۲	انگیزه پژوهش
۳-۱	ساختار پایان نامه
۵	فصل ۲: مروری بر مفاهیم
۵-۲	مقدمه
۵-۲-۲	مجازی سازی توابع شبکه
۷-۲-۲	ساختار NFV
۹-۲-۲	جایابی توابع مجازی شبکه
۱۱-۲	شبکه نرم افزار محور
۱۲-۳-۲	پروتکل Openflow
۱۴-۲	رابطه بین SDN و NFV
۱۶-۲	مسیریابی
۱۸-۵-۲	SDN در مسیریابی
۲۰-۲	روش های اکتشافی
۲۲-۲	پیشینه پژوهش
۲۵	فصل ۳: هماهنگ سازی، مسیریابی و جایابی VNF

۱-۳	مقدمه	۲۵
۲-۳	ساختار کلی سیستم	۲۵
۳-۳	بیان مساله	۲۸
۴-۳	فرمول بندی بهینه سازی خطی و مختلط عدد صحیح	۳۱

فصل ۴: روش های پیشنهادی و شبیه سازی ها

۱-۴	مقدمه	۳۵
۲-۴	مساله بسته بندی در جعبه ها	۳۵
۱-۲-۴	روش های حل BPP	۳۶
۲-۲-۴	BPP با اندازه های متفاوت	۴۱
۳-۴	مساله یافتن کوتاه ترین مسیر	۴۳
۴-۴	روش های اکتشافی پیشنهادی	۴۵
۱-۴-۴	روش SSP محور	۴۶
۲-۴-۴	روش BFD تطبیقی	۴۸
۵-۴	شبیه سازی و نتایج	۴۹
۱-۵-۴	مفروضات شبیه سازی	۴۹
۲-۵-۴	نتایج شبیه سازی	۵۰

فصل ۵: جمع بندی و پیشنهادها

۱-۵	مقدمه	۵۹
۲-۵	نتیجه گیری	۵۹
۳-۵	پیشنهادهای	۶۰

مراجع

پیوست آ:	روش مجموع زیر مجموعه	۶۵
آ-۱	روش عقب گرد	۶۶

فهرست تصاویر

۱-۱	بهبودهای عملکردی در نسل پنجم [۸]	۲
۱-۲	ساختار NFV [۲۸]	۷
۲-۲	دو الگوی مختلف از قرار دهی VNF برای یک زنجیره سرویس یکسان [۲۰]	۱۰
۳-۲	تفاوت معماری شبکه های مرسوم با SDN [۳۷]	۱۳
۴-۲	مقایسه مفاهیم شبکه نرم افزار محور و مجازی سازی توابع شبکه [۳۷]	۱۵
۵-۲	ساختار NFV مبتنی بر SDN [۲۶]	۱۶
۶-۲	نمودار اولر برای کلاس های مختلف پیچیدگی [۳]	۲۱
۱-۳	ساختار کلی سیستم [۲۵]	۲۷
۲-۳	تطبیق ساختار سیستم با ETSI [۲۵]	۲۹
۱-۴	شبیه کد روش اولین برازش [۳۵]	۳۷
۲-۴	شبیه کد روش اولین برازش کاهشی [۳۵]	۳۷
۳-۴	شبیه کد روش بهترین برازش [۳۵]	۳۸
۴-۴	شبیه کد روش بهترین برازش کاهشی [۳۵]	۳۹
۵-۴	شبیه کد روش برازش بعدی [۳۵]	۳۹
۶-۴	شبیه کد روش برازش بعدی کاهشی [۳۵]	۴۰
۷-۴	شبیه کد روش باقی مانده بیشینه [۳۵]	۴۰
۸-۴	شبیه کد روش باقی مانده بیشینه با صف اولویت بندی [۳۵]	۴۱
۹-۴	شبیه کد الگوریتم دایجسترا [۱۵]	۴۵
۱۰-۴	شبیه کد الگوریتم دایجسترا [۱۵]	۴۷

۵۲	۱۱-۴ شبه کد الگوریتم دایجسترا [۱۵]
۵۳	۱۲-۴ گراف توپولوژی شبکه
۵۴	۱۳-۴ میانگین هزینه تخصیص زنجیره خدمات بر حسب تعداد زنجیره‌های خدمات
۵۵	۱۴-۴ میانگین هزینه تخصیص زنجیره خدمات بر حسب اندازه بافر لینک‌ها
۵۶	۱۵-۴ میانگین تعداد سرورهای تخصیص یافته
۵۷	۱۶-۴ زمان لازم جهت اجرای الگوریتم‌ها
۶۷	۱-آ شبه کد الگوریتم عقب‌گرد

فصل ۱

مقدمه

۱-۱ مقدمه

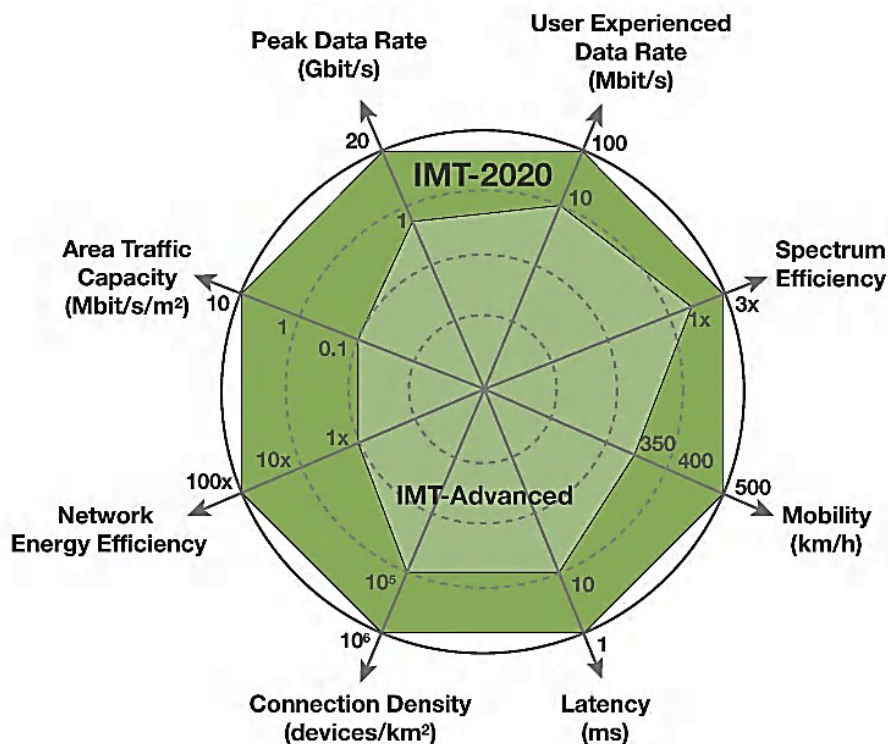
نسل پنجم فناوری موبایل، به منظور پاسخگویی به نیازها و ضروریات تجاری و اجتماعی فعلی مطرح شده است. پاسخگویی به این ضروریات، نیازمند بهبود عملکرد شبکه در زمینه‌هایی چون نرخ داده، تاخیر، قابلیت اطمینان، حرکت‌پذیری و غیره است. در شکل ۱-۱ برخی بهبودهای ایجادشده در عملکرد شبکه نسل پنجم نسبت به نسل پیشین از دیدگاه اتحادیه بین‌المللی مخابرات (ITU) مشاهده می‌شود.

۲-۱ انگیزه پژوهش

اجرای طراحی، مدیریت و بهره‌برداری از زیر ساخت شبکه نیازمند تکامل به کمک فناوری‌ها و معماری‌های نوین است. به کمک زمینه‌های پژوهشی نوظهور مانند مجازی‌سازی توابع شبکه یا NFV و شبکه نرم‌افزار محور یا SDN، انعطاف‌پذیری بیشتری برای شبکه‌ها فراهم می‌گردد.

این انعطاف‌پذیری بالاتر، باعث به وجود آمدن چالش‌هایی پیرامون مبحث پیاده‌سازی و جایابی بهینه VNF ها و مسیریابی جریان از میان آن‌ها می‌شود. برای مثال کنترل هزینه، کیفیت سرویس^۱ یا QoS، تاخیر، ازدحام،

¹Quality Of Service



شکل ۱-۱: بهبودهای عملکردی در نسل پنجم [۸]

انرژی مصرفی، عدالت و ... از جمله چالش هایی است که در این مبحث مطرح هستند و به عنوان قید در مساله بهینه سازی جایابی VNF ها و مسیریابی مورد توجه قرار می گیرند. بنابراین ادغام NFV و SDN با ارائه یک چارچوب کارآمد برای کنترل همکارانه و برنامه ریزی توابع شبکه، این امکان را فراهم می سازد تا اعمال نوآوری در شبکه ساده تر گردد. هدف این پایان نامه بررسی همزمان جایابی توابع شبکه مجازی و حل مساله مسیریابی با کمک SDN می باشد. حل توام این دو مساله هماهنگ سازی^۲ نامیده می شود.

۳-۱ ساختار پایان نامه

همانطور که توضیح داده شد در این پروژه قصد داریم مدیریت مسیریابی در شبکه و جایابی VNF ها به کمک یک ساختار نرم افزار محور را بررسی کنیم و با در نظر گرفتن آگاهی ها و اطلاعات موجود، سعی بر تحلیل عملکرد

²orchestration

سیستم نماییم.

در فصل دوم این گزارش، ابتدا مروري بر مفاهيم پایه مطرح شده در نسل پنجم شبکه‌های مخابراتی صورت می‌گیرد. در فصل سوم، مسئله مدیریت جایابی توابع مجازی شبکه و مسیریابی توسط SDN در شبکه تعریف شده و راه‌حل‌های موجود آن مورد بررسی قرار می‌گیرند. فصل چهارم به بررسی الگوریتم‌های اکتشافی موجود می‌پردازد و دوروش اکتشافی برای مساله بهینه‌سازی مطرح شده ارائه می‌شود و عملکرد یکی از آنها به کمک اجرای شبیه‌سازی‌های مختلف، با حالت بهینه مقایسه می‌گردد. در نهایت با جمع‌بندی نتایج و ارائه پیشنهادها در فصل پنجم، گزارش به اتمام می‌رسد.

فصل ۲

مروری بر مفاهیم

۱-۲ مقدمه

هدف از این فصل، معرفی مفاهیم و فناوری‌هایی است که پایه‌های فصول بعدی را تشکیل می‌دهند. در ادامه به طور خلاصه دو مفهوم شبکه نرم‌افزار محور و مجازی‌سازی توابع شبکه که از مهم‌ترین نوآوری‌های ارائه شده در نسل پنجم هستند، مورد بررسی قرار می‌گیرند. سپس مفاهیم مسیریابی و الگوریتم‌های اکتشافی که در فصول بعد از آن‌ها استفاده می‌شود، معرفی می‌گردند.

۲-۲ مجازی‌سازی توابع شبکه

ارائه خدمات در صنعت ارتباطات راه دور به‌طور سنتی بر مبنای اپراتورهای شبکه است که تجهیزات و وسایل فیزیکی اختصاصی را برای هر عملکردی که بخشی از یک سرویس خاص است، بکار می‌گیرند. علاوه بر این، اجزای سرویس دارای ترتیب دقیقی هستند که باید در توپولوژی شبکه و در قرار دهی اجزا سرویس انعکاس داده شود. این امر، همراه با الزامات برای کیفیت بالا، پایداری و پیروی دقیق از پروتکل، منجر به چرخه تولید طولانی، تغییر خدمات بسیار کم و وابستگی شدید به سخت‌افزار تخصصی شده است. همچنین نیازهای کاربران برای خدمات متنوع و جدید (کوتاه‌مدت) با نرخ داده بالا همچنان افزایش می‌یابد.

بنابراین، ارائه‌دهندگان خدمات مخابراتی^۱ یا TSP ها باید به‌طور مداوم تجهیزات فیزیکی جدید خریده، ذخیره و استفاده کنند. این حالت نه تنها نیاز به مهارت‌های بالا و سریع در حال تغییر برای تکنسین‌های استفاده و مدیریت کننده این تجهیزات دارد، بلکه نیاز به توسعه انبوه تجهیزات شبکه مانند ایستگاه‌های پایه^۲ دارد. این‌ها همه منجر به هزینه‌های راه‌اندازی و نگهداری بالا می‌شود.

علاوه بر این، حتی با توجه به این خواسته‌های بالای مشتریان، افزایش اعمال‌شده در هزینه‌های اولیه و عملیاتی نمی‌تواند منجر به آونمان‌های بالاتر شود، زیرا ارائه‌دهندگان خدمات متوجه شده‌اند که با توجه به وجود رقابت زیاد بین بقیه ارائه‌دهندگان و همچنین خدماتی که بر روی کانال‌های داده آن‌ها ارائه می‌شوند، افزایش قیمت‌ها تنها منجر به کاهش مصرف‌کنندگان می‌شود. بنابراین، TSP ها مجبور به پیدا کردن راه‌هایی برای ساختن شبکه‌های پویا و آگاه به سرویس هستند تا چرخه‌های تولید، هزینه‌های عملیاتی و اولیه را کاهش داده و سرعت تغییر خدمات را افزایش دهند.

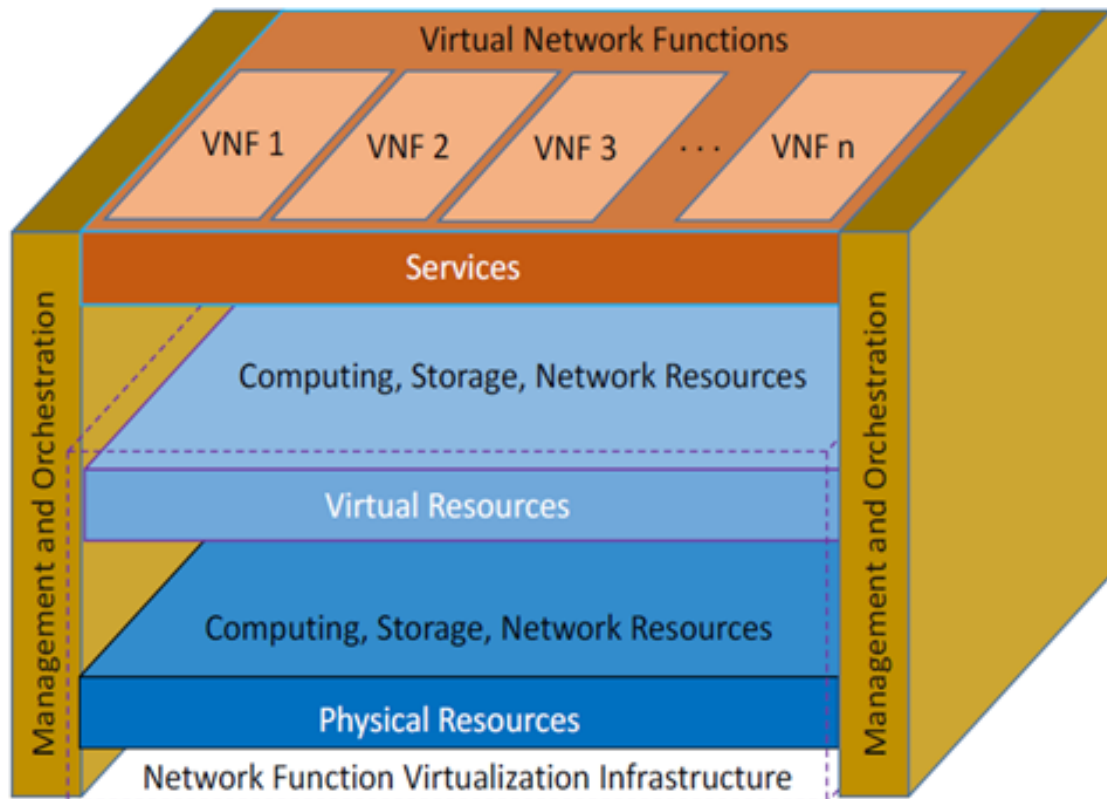
NFV به‌عنوان راهی برای حل این چالش‌ها با استفاده از فناوری مجازی‌سازی پیشنهاد شده است تا راهی جدید برای طراحی، راه‌اندازی و مدیریت خدمات شبکه ارائه دهد.

مفهوم و کار مشارکتی روی NFV در اکتبر ۲۰۱۲ شکل گرفت، هنگامی که تعدادی از TSP های پیشرو در جهان به‌طور مشترک یک مقاله تهیه کرده و خواستار اقدام صنعتی و تحقیقاتی شدند. در نوامبر ۲۰۱۲، اپراتورها موسسه استاندارد ارتباطات اروپایی^۳ یا ETSI را به‌عنوان مرکز گروه مشخصات صنعتی NFV انتخاب کردند. ایده اصلی NFV جدا کردن تجهیزات شبکه فیزیکی از توابعی که بر روی آن‌ها اجرا می‌شود است. این به این معنی است که یک تابع شبکه مانند یک فایروال می‌تواند به‌عنوان نمونه‌ای از نرم‌افزار ساده به TSP ارسال شود. این امر اجازه تثبیت بسیاری از انواع تجهیزات شبکه بر روی سرورهای بزرگ، سوئیچ‌ها و انبارها را می‌دهد که می‌توانند در مراکز داده، گره‌های توزیع شده شبکه و در انتهای سمت کاربر قرار گرفته باشند. به این ترتیب، یک سرویس داده می‌تواند به یک مجموعه از توابع مجازی شده شبکه یا VNF ها تجزیه شود و پس از آن می‌تواند روی نرم‌افزار در حال اجرا بر روی یک یا چند سرور فیزیکی استاندارد صنعتی اجرا شود. پس از آن VNF ها می‌توانند در مکان‌های مختلف شبکه (به‌عنوان مثال برای معرفی یک سرویس که مشتریان در یک مکان جغرافیایی خاص را هدف قرار داده)، بدون نیاز به خرید و نصب سخت‌افزار جدید، جایگزین شوند. [۱۱]

¹Telecommunication Service Providers

²Base Station

³European Telecommunications Standards Institute



شکل ۲-۱: ساختار NFV [۲۸]

۱-۲-۲ ساختار NFV

طبق ETSI معماری NFV شامل سه عنصر کلیدی است: زیرساخت مجازی سازی توابع شبکه، توابع مجازی شبکه و مدیریت و هماهنگ سازی NFV. که در این بخش این عناصر تعریف می شوند. که در شکل ۱-۲-۲ نشان داده شده است.

- زیرساخت NFV یا NFVI^۴: NFVI ترکیبی از منابع سخت افزاری و نرم افزاری است که محیطی را که در آن VNF ها مستقر هستند تشکیل می دهند. منابع فیزیکی شامل سخت افزار محاسباتی تجاری تولید انبوه (COTS)، انبار و شبکه (شامل گره و لینک) است که پردازش، ذخیره سازی و اتصال به VNF ها را فراهم می کند. منابع مجازی، انتزاعی از محاسبات، ذخیره سازی و منابع شبکه هستند. انتزاع با استفاده از یک لایه مجازی سازی (بر اساس یک ناظر ماشین مجازی) به دست می آید، که منابع مجازی را از منابع

⁴NFV Infrastructure

فیزیکی جدا می‌کند. در محیط مرکز داده، محاسبات و منابع ذخیره‌سازی ممکن است به صورت یک یا چند دستگاه مجازی^۵ یا VM نمایش داده شوند، در حالی که شبکه‌های مجازی از پیوندها و گره‌های مجاز ساخته شده‌اند. یک گره مجازی یک جزء نرم‌افزاری با قابلیت میزبانی یا مسیریابی است. لینک مجازی یک اتصال منطقی دو گره مجازی است که به نظر آن‌ها مثل یک پیوند فیزیکی مستقیم با خواص در حال تغییر دینامیکی ظاهر می‌شود. [۲۸]

• **توابع و خدمات شبکه مجازی:** یک تابع شبکه یک بلوک عملکردی در زیرساخت شبکه است که رابط‌های خارجی و رفتار عملکردی مشخص دارد [۱۱]. نمونه‌هایی از NF ها، عناصر یک شبکه خانگی هستند، مثل ورودی محلی^۶؛ و توابع معمولی شبکه، مثل سرورهای DHCP، فایروال‌ها، و غیره. بنابراین یک VNF یک پیاده‌سازی از یک NF است که روی منابع مجازی مثل یک VM مستقر شده است. یک VNF تنها، ممکن است از اجزای داخلی چندگانه تشکیل شده باشد و از این رو می‌توان آن را در چند VM مستقر کرد، در این صورت هر VM یک جزء واحد از VNF را میزبانی می‌کند [۲۸]. یک سرویس، یک ارائه فراهم شده توسط TSP است که از یک یا چند NF تشکیل شده است. در مورد NF NFV ها، که سرویس را تشکیل می‌دهند، مجازی‌سازی می‌شوند و روی منابع مجازی مانند VM اجرا می‌شوند. با این حال، در نظر کاربران، سرویس‌ها (چه بر اساس توابع در حال اجرا روی تجهیزات اختصاصی یا روی VM ها) باید عملکرد مشابهی داشته باشند. تعداد، نوع و سفارش VNF ها که سرویس را تشکیل می‌دهند، توسط ویژگی‌های عملکردی و رفتاری سرویس تعیین می‌شود. بنابراین، رفتار سرویس وابسته به VNF های تشکیل دهنده است.

• **مدیریت و هماهنگ‌سازی NFV:** با توجه به چارچوب ETSI، MANO قابلیت‌هایی را برای تهیه VNF ها و عملیات مربوطه مانند پیکربندی^۷ VNF ها و زیرساخت این عملکردها فراهم می‌کند. این قابلیت‌ها، شامل هماهنگ‌سازی و مدیریت چرخه عمر منابع فیزیکی و یا نرم‌افزاری است که از مجازی‌سازی زیرساخت و مدیریت چرخه عمر VNF ها پشتیبانی می‌کند. همچنین شامل پایگاه‌های داده‌ای^۸ است که برای ذخیره اطلاعات و مدل‌های داده‌ای استفاده می‌شود که هم استقرار و هم خواص چرخه عمر توابع، خدمات و منابع را تعریف می‌کنند. MANO NFV بر روی تمام وظایف مدیریتی

^۵Virtual Machine

^۶residential gateway

^۷configuration

^۸databases

اختصاصی مجازی سازی که در چارچوب NNF ضروری است تمرکز می کند. علاوه بر این، این چارچوب رابط‌هایی را تعریف می کند که می توانند برای برقراری ارتباط بین اجزای مختلف MANO NNF و همچنین هماهنگی با سیستم‌های مدیریت شبکه سنتی مانند سیستم پشتیبانی عملیات^۹ یا OSS و سیستم‌های پشتیبانی تجاری^{۱۰} یا BSS استفاده شوند تا اجازه مدیریت هم VNF ها و هم توابع در حال اجرا بر روی تجهیزات سابق را بدهند.

۲-۲-۲ جایابی توابع مجازی شبکه

توابع شبکه که همچنین تحت عنوان جعبه‌های میانی^{۱۱} شناخته می شوند، نقش مهمی در شبکه‌های مدرن، از جمله شبکه‌های تلفن همراه، شبکه‌های سازمانی و شبکه‌های مرکز داده بازی می کنند. توابع شبکه عملکرد شبکه را بهبود می بخشد (مانند بهینه سازی، WAN، پروکسی وب و تبدیل کننده ویدئویی، متعادل کننده بار)، امنیت را بهبود می بخشد (مانند فایروال، IDS/IPS) و یا بر ترافیک نظارت می کنند (مانند رهگیری قانونی، نظارت بر شبکه غیرفعال).

جایابی VNF مساله انتخاب مجموعه مکان‌های بهینه برای زنجیره‌ای از VNF ها، مطابق درخواست سرویس و ویژگی‌های فعلی منابع محاسباتی موجود و لینک‌های شبکه است. جایابی توابع مجازی شبکه می تواند بر اساس تقاضای آن‌ها در شبکه مدیریت شود که این تقاضا با گذشت زمان به صورت پویا تغییر می کند. جهت اعمال سیاست‌ها و یا ارائه خدمات شبکه، ترافیک باید از میان دنباله خاصی از VNF ها که زنجیره سرویس نامیده می شود، عبور کند. زنجیره کردن سرویس به روش‌هایی جهت ترتیب دادن به دنباله VNF ها و اجرای آن‌ها اشاره دارد که در آن‌ها ترافیک باید از میان زنجیره سرویس مناسب عبور کند. بهینه سازی جایابی VNF هایی که به صورت پویا ایجاد شده اند و تعیین بهترین مسیر برای حرکت VNF ها در زنجیره سرویس، مسائل چالش برانگیزی هستند [۲۰].

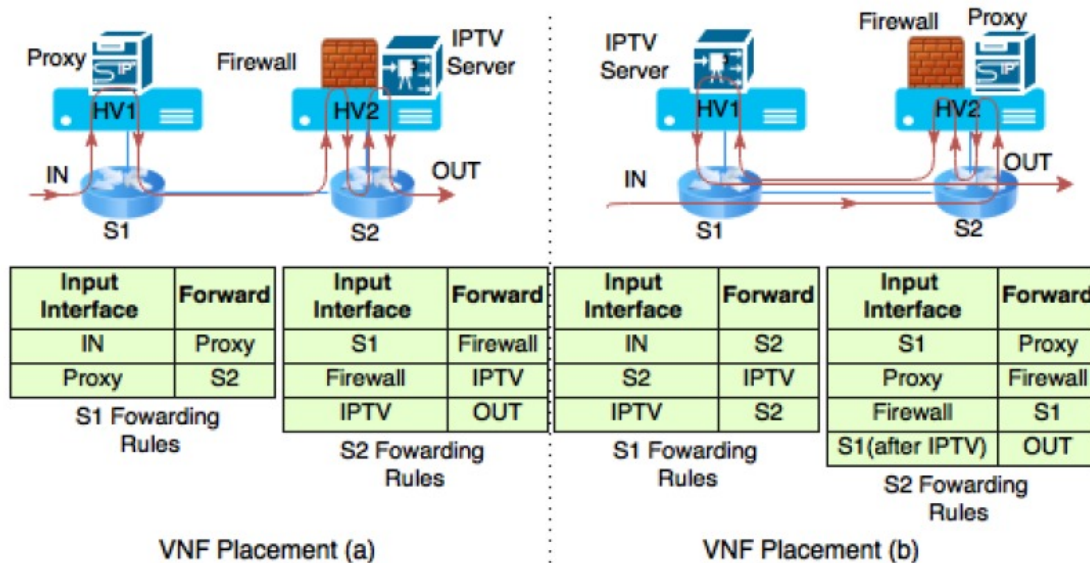
جایابی VNF ها دارای اثرگذاری گسترده روی ابعاد مختلف شبکه است، از جمله:

- به طور قابل توجهی بر بار روی سوئیچ‌ها تأثیر می گذارد.
- کارایی استفاده از پهنای باند را تعیین می کند.

^۹Operations Support System

^{۱۰}Business Support System

^{۱۱}middleboxes



شکل ۲-۲: دو الگوی مختلف از قرار دهی VNF برای یک زنجیره سرویس یکسان [۲۰]

- میزان هزینه‌های اولیه و عملیاتی را تحت تأثیر قرار می‌دهد.
- چابکی خدمات ارائه‌شده را تحت تأثیر قرار می‌دهد.
- روی تأخیر انتها به انتها تأثیر می‌گذارد.

شکل ۲-۲-۲، یک نمونه از جایابی VNF در زنجیره سرویس در یک شبکه را نشان می‌دهد. S1 و S2 جریان متناظر را از طریق زنجیره سرویس که شامل پروکسی، فایروال و سرور IPTV است، ارسال می‌کنند. در مقابل، قراردهی VNF در شکل b یک روش دیگر را نشان می‌دهد که دارای قوانین ارسال بیشتری است و مسیر طولانی‌تری باید توسط جریان پیموده شود. یک تغییر جزئی در توالی جایابی VNF می‌تواند یک ناکارآمدی بزرگ در مدیریت ترافیک بین VNF ها، یعنی پینگ‌پونگ ترافیک، اعمال کند و منجر به هدر رفتن منابع شبکه مانند حافظه سوئیچ (TCAM) و پهنای باند شبکه شود. برای یک شبکه بزرگ با تعداد زیادی از زنجیره‌های سرویس، این مشکل قابل توجه خواهد شد. حتی یک مرکز داده کوچک نیز میلیون‌ها جریان دارد.

قرار دهی VNF و تعبیه شبکه می‌تواند به عنوان یک مساله بهینه‌سازی ریاضی مطرح شود، که به دنبال بهینه‌سازی یک تابع هدف است. این تابع هدف در راستای تأمین هدف ادمین مرکز داده است و درعین حال، مجموعه‌ای از محدودیت‌ها مثل محدودیت‌های فنی محاسباتی، زیرساخت‌های شبکه و نیازهای کاربران را نیز در نظر می‌گیرد.

۳-۲ شبکه نرم افزار محور

راه اندازی شبکه های مخابراتی پیشین، نیازمند فناوری های نرم افزاری و شبکه های سخت افزار محور مختلف مثل پروتکل ها، سوئیچ ها و مسیریاب ها بود. بنابراین شبکه های ارتباطی بی سیم و سیمی برای داشتن تعداد زیادی دستگاه متصل به شبکه با مشخصات و ویژگی های مختلف، با مشکلات پیچیده و اساسی مواجه می شدند. به علاوه، دستگاه ها در شبکه های داده ی سنتی، نیازمند پیکربندی جداگانه بودند، به روزرسانی وظایف دستگاه ها زمان بر بود و ایجاد تحول در عملکردهای سخت افزار فقط توسط سازنده ممکن بود. در نتیجه اجرای یک سیاست جدید در شبکه، نیازمند پیکربندی چندین دستگاه بوده و فرآیند بازپیکربندی سخت و کند بود.

در این شبکه ها، دستگاه هایی مثل سوئیچ و مسیریاب وظایف خاصی را انجام می دادند. نتیجتاً این شبکه ها دارای معماری ثابت بوده و عملکردهای شبکه به کندی تحول پذیر بودند. از سوی دیگر، استفاده از سرویس های مختلف روی گوشی های تلفن همراه، سرورهای مجازی و سرویس های ابری به شدت افزایش داشته که همه این فعالیت ها به ساختاری پویا برای شبکه نیاز دارند.

از دیگر سو، در دهه گذشته الگوی ترافیک تغییر کرده است و کاربران می خواهند به کاربردها، زیرساخت ها و سایر منابع فناوری اطلاعات دسترسی داشته باشند. همچنین، شبکه های دسترسی و داده در حال رشد هستند و درخواست برقراری ارتباط در حال افزایش است که به معنی افزایش پهنای باند و راه اندازی اجزای جدید زیرساخت است. کنترل مجموعه داده های حجیم^{۱۲} برای شبکه های سنتی بسیار پرهزینه بود.

همچنین عملکردهای شبکه، محدود و وابسته به تولیدکنندگان سخت افزار بود که این مورد نیز از جمله مشکلات صاحبان شبکه بود. با افزایش اندازه شبکه، پیچیدگی آن نیز رشد می کند. پس زیرساخت شبکه به عملیات های پویا و انعطاف پذیرتر، برنامه پذیر بودن و دستگاه های اصلاح پذیر نیاز دارد. در حال حاضر، بهترین فناوری لازم برای رسیدن به رفتار مذکور شبکه نرم افزار محور (SDN) است. SDN یک کنترل مرکزی در شبکه فراهم می آورد که در آن لایه داده^{۱۳} از لایه کنترل^{۱۴} جدا شده است. اگرچه اصطلاح SDN در دهه ی پیش ابداع شده است، ولی مفهوم SDN با ایده های به دست آمده از فناوری های مختلف دیگر از سال ۱۹۹۰ تکامل یافته است. ایده ی جداسازی^{۱۵} صفحه داده^{۱۶} و صفحه کنترل^{۱۷} از نقاط کنترل شبکه پیاده سازی شده در شبکه های تلفن، به دست

¹²big data

¹³data layer

¹⁴control layer

¹⁵decoupling

¹⁶data plane

¹⁷control plane

آمده است. [۳۴] جداسازی لایه‌ی کنترل از لایه‌ی داده توانایی برنامه‌پذیری، بهبود عملکرد و مدیریت از راه دور زیرساخت را به شبکه اضافه می‌کند. این ساختار به شبکه و کاربردهای تجاری اجازه می‌دهد تا سیاست‌های شبکه را طبق تغییرات کاربران و برنامه‌ها تغییر دهند.

در SDN با متمرکز شدن فعالیت‌های کنترلی و همچنین جداسازی زیرساخت شبکه از کاربردها، امکان مدیریت بهینه‌ی زیرساخت ایجاد می‌شود. دستگاه‌های شبکه، به جای داشتن قابلیت فهم و تفسیر پروتکل‌های مختلف، صرفاً نیاز دارند دستورالعمل‌ها را از کنترل‌کننده SDN دریافت کنند. پس هر المان شبکه می‌تواند به صورت آنی تغییر کند. همچنین، شبکه می‌تواند نیازهای فوری تجاری را تأمین کرده و به شخصی‌سازی شبکه کمک کند. شکل ۲-۳ تفاوت معماری شبکه‌های مرسوم و SDN را نمایش می‌دهد. در شبکه‌های مرسوم، صفحه کنترل و صفحه داده به صورت یکپارچه در هر دستگاه شبکه وجود دارد. برای توانایی برنامه‌پذیری، بهبود عملکرد، توانایی کنترل از راه دور و مدیریت میان زیرساخت‌ها، SDN روی چهار ویژگی تمرکز کرده است:

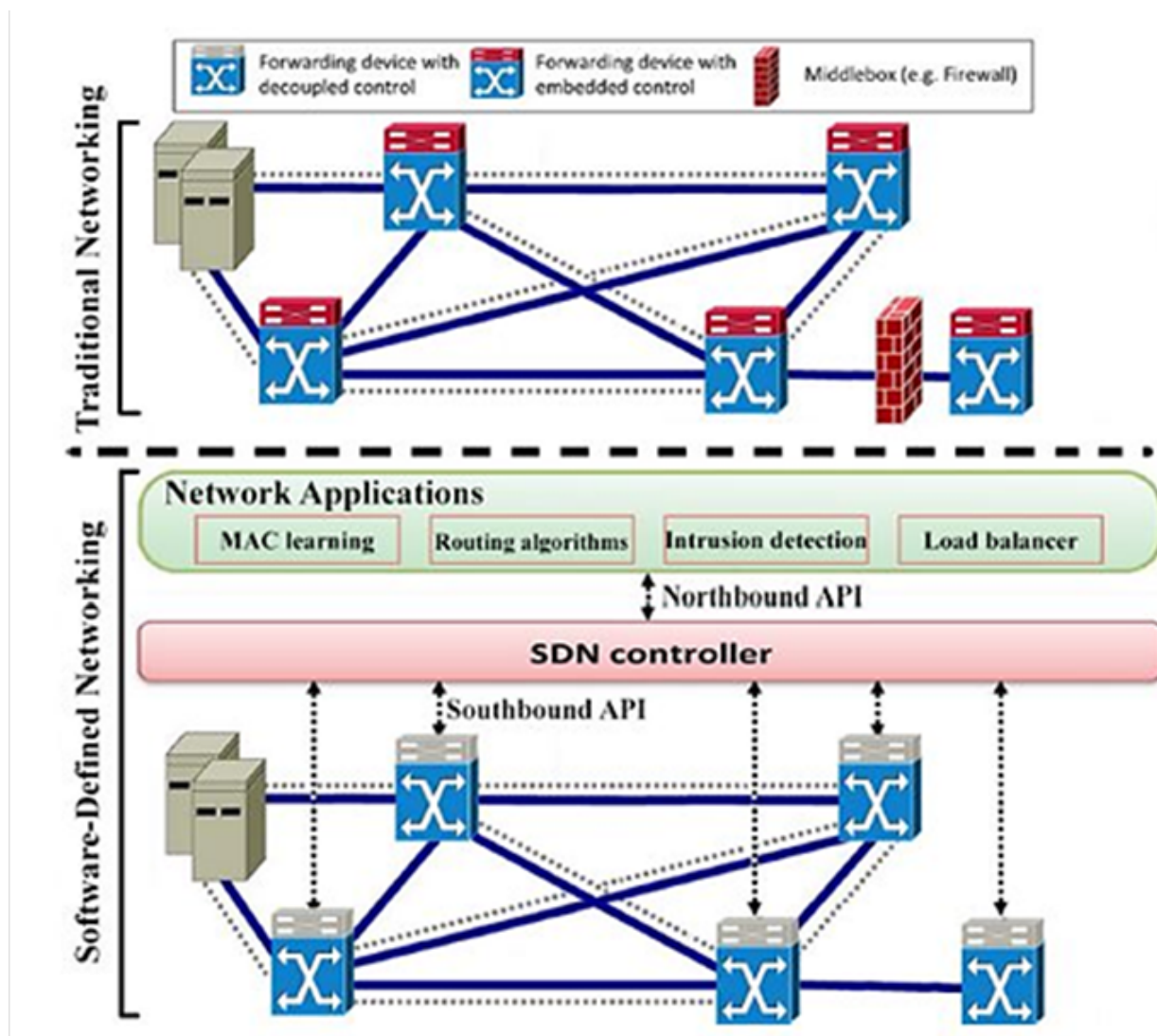
- جدا کردن صفحه داده از صفحه کنترل
- مشاهده جزئیات شبکه با کنترل‌کننده مرکزی شده
- رابط باز^{۱۸} بین دستگاه‌های صفحه کنترل (کنترل‌کننده‌ها) و دستگاه‌ها در لایه داده
- قابلیت برنامه‌پذیری شبکه با استفاده از کاربردهای خارجی^{۱۹} [۳۷]

۱-۳-۲ پروتکل Openflow

می‌توان تولد SDN را با معرفی OpenFlow در ۲۰۰۸ هم‌زمان دانست زیرا تا پیش از آن، به دلیل فقدان زیرساخت مناسب و پشتیبانی سخت‌افزار، مدل‌های ارائه‌شده برای SDN نمی‌توانستند پیاده‌سازی شوند. در SDN ارتباط میان زیرساخت و لایه‌ی کنترل با پروتکل OpenFlow برقرار می‌شود. اگرچه اصطلاح SDN و OpenFlow به هم مرتبط است ولی نمی‌توان از آن‌ها به جای یکدیگر استفاده کرد. SDN نوعی معماری شبکه است که به شبکه توانایی برنامه‌پذیری می‌دهد. ولی OpenFlow یک پروتکل است که برای ارتباط بین سوئیچ‌ها و کنترل‌کننده SDN روی سوئیچ‌ها پی‌کربندی می‌شود. OpenFlow به کنترل‌کننده SDN اجازه تعیین

¹⁸open interface

¹⁹external applications



شکل ۲-۳: تفاوت معماری شبکه های مرسوم با SDN [۳۷]

مسیر داده در میان سوئیچ‌های شبکه را می‌دهد. کنترل‌کننده باید بتواند با سایر کنترل‌کننده‌ها ارتباط برقرار کرده و شبکه را به صورت پویا و مطابق با نیازهای آن کنترل کند.

۴-۲ رابطه بین SDN و NFV

NFV و SDN دارای اشتراکات زیادی هستند، زیرا هر دو از حرکت به سمت نرم‌افزار باز و سخت‌افزار شبکه استاندارد حمایت می‌کنند. به طور خاص، همان‌طور که NFV قصد دارد NF ها را بر روی سخت‌افزار استاندارد صنعتی اجرا کند، لایه کنترل SDN می‌تواند به عنوان صرف نرم‌افزار در حال اجرا بر روی سخت‌افزار استاندارد صنعتی پیاده‌سازی شود. علاوه بر این، هم NFV و هم SDN به دنبال دستیابی به اتوماسیون و مجازی‌سازی تا به اهداف مربوطه خود دست یابند. در حقیقت، NFV و SDN ممکن است تا حد زیادی تکمیل‌کننده هم باشند و از این رو ترکیب آن‌ها در یک راه‌حل شبکه ممکن است به ارزش بیشتر منجر شود.

برای مثال، اگر یک کنترل‌کننده SDN قادر به اجرای روی یک VM باشد، ممکن است به عنوان بخشی از یک زنجیره سرویس به کار رود. این بدان معنی است که برنامه‌های کنترل و مدیریت مرکزی شده (مانند تعادل بار، نظارت و تجزیه و تحلیل ترافیک) که در SDN استفاده می‌شود، تا حدودی می‌تواند به عنوان VNF تفسیر شوند و از این رو، از ویژگی‌های قابلیت اطمینان و انعطاف‌پذیری NFV بهره‌مند شوند. به همان شیوه، SDN می‌تواند راه‌اندازی NFV را با ارائه یک روش انعطاف‌پذیر و خودکار از زنجیره کردن توابع، تهیه و تنظیم اتصالات شبکه و پهنای باند، اتوماسیون عملیات، امنیت و کنترل خط مشی سرعت بخشد.

با این حال، SDN و NFV مفاهیم متفاوتی با هدف بررسی جنبه‌های مختلف یک راه‌حل شبکه مبتنی بر نرم‌افزار هستند. NFV قصد دارد NF ها را از عناصر سخت‌افزاری اختصاصی جدا کند، در حالی که SDN بر جداسازی دست‌کاری بسته‌ها و اتصالات از کنترل کلی شبکه تمرکز دارد. به عبارت دیگر، مفهوم NFV از مفهوم مجازی‌سازی که در معماری SDN استفاده شده است، متفاوت است. در معماری، SDN مجازی‌سازی، تخصیص منابع انتزاعی به مشتریان یا برنامه‌های خاص است، در NFV هدف این است که NF ها را از سخت‌افزار اختصاصی جدا کرد، به عنوان مثال اجازه می‌دهد تا آن‌ها روی پلت فرم سرور در مراکز داده ابری میزبانی شوند.

رابطه بین SDN و NFV در شکل ۴-۲ خلاصه شده است. [۳۷]

همچنین مقایسه مفاهیم شبکه نرم‌افزار محور و مجازی‌سازی توابع شبکه سیستم NFV نرم‌افزار محور در

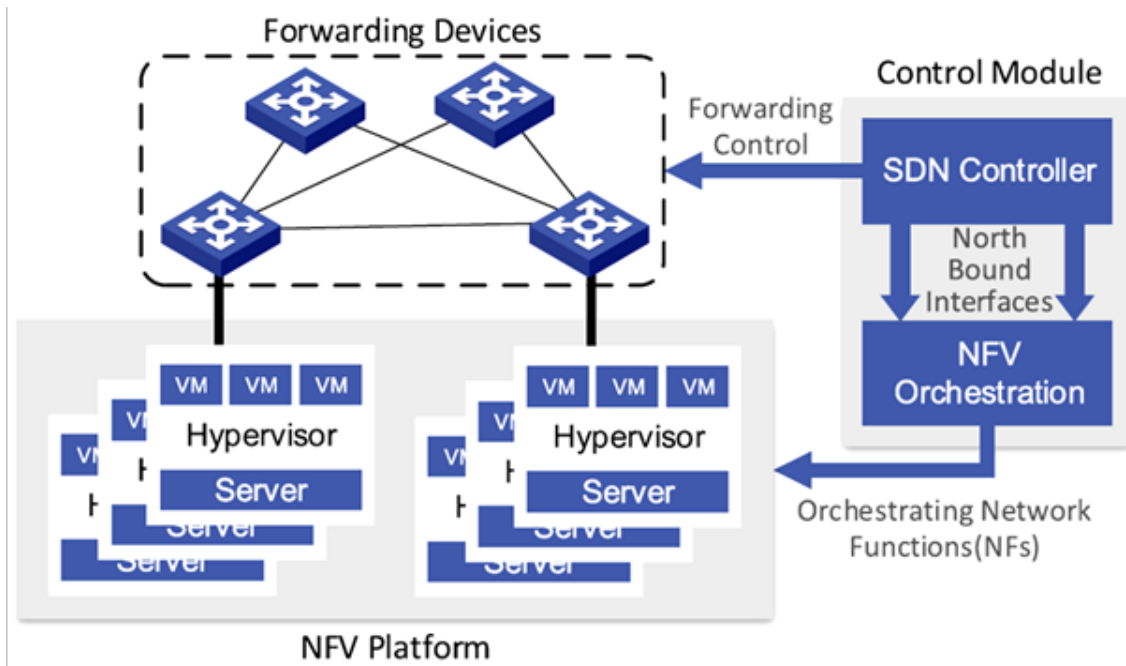
Issue	NFV (Telecom Networks)	Software Defined Networking
Approach	Service/Function Abstraction	Networking Abstraction
Formalization	ETSI	ONF
Advantage	Promises to bring flexibility and cost reduction	Promises to bring unified programmable control and open interfaces
Protocol	Multiple control protocols (e.g. SNMP, NETCONF)	OpenFlow is de-facto standard
Applications run	Commodity servers and switches	Commodity servers for control plane and possibility for specialized hardware for data plane
Leaders	Mainly Telecom service providers	Mainly networking software and hardware vendors
Business Initiator	Telecom service providers	Born on the campus, matured in the data center

شکل ۲-۴: مقایسه مفاهیم شبکه نرم‌افزار محور و مجازی‌سازی توابع شبکه [۳۷]

شکل ۲-۴ نشان داده شده است. این ساختار شامل یک ماژول کنترل، دستگاه‌های ارسال و پلت فرم NFV در لبه شبکه است. منطق ارسال بسته توسط کنترل‌کننده SDN تعیین می‌شود و از طریق جداول ارسال، روی دستگاه‌های ارسال^{۲۰} پیاده‌سازی می‌شوند. پروتکل‌های کارآمد، مانند Openflow، می‌توانند به عنوان رابط‌های استاندارد در برقراری ارتباط بین کنترل‌کننده مرکزی و دستگاه‌های ارسال گسترده مورد استفاده قرار گیرند.

پلت فرم NFV، از سرورهای مناسب برای پیاده‌سازی NF ها با پهنای باند بالا و قیمت پایین استفاده می‌کند. ناظرهای ماشین‌های مجازی روی سرورها اجرا می‌شوند تا از ماشین‌های مجازی‌ای که توابع شبکه‌ها را اجرا می‌کنند پشتیبانی کنند. این پلت فرم امکان وجود توابع پردازشی قابل تنظیم و قابل برنامه‌ریزی لایه داده را فراهم می‌کند. از جمله این توابع، جعبه میانی فایروال، IDS ها و پروکسی‌ها هستند که به عنوان نرم‌افزار روی ماشین‌های مجازی اجرا می‌شوند. توابع شبکه به صورت نرم‌افزاری خالص به اپراتور شبکه تحویل داده می‌شوند. کنترل‌کننده SDN و سیستم هماهنگ‌سازی NFV ماژول کنترل منطقی را تشکیل می‌دهند. سیستم هماهنگ‌سازی NFV مسئول تهیه توابع شبکه مجازی است و از طریق رابط‌های استاندارد توسط کنترل‌کننده SDN کنترل می‌شود. پس از فراهم کردن توپولوژی شبکه و الزامات سیاست، ماژول کنترل تخصیص بهینه تابع (اختصاص دادن توابع شبکه به بعضی از VM ها) را محاسبه می‌کند و مشخصات سیاست منطق را به مسیرهای بهینه مسیریابی تبدیل می‌کند. پس از به دست آوردن توپولوژی شبکه و الزامات سیاست، ماژول کنترل تخصیص تابع بهینه (اختصاص دادن کارکردهای شبکه به بعضی از VM ها) را محاسبه می‌کند تخصیص تابع توسط سیستم هماهنگ‌سازی NFV اعمال می‌شود و کنترل‌کننده با نصب قوانین ارسال روی دنباله‌های مورد نیاز و مناسب از VM ها و دستگاه‌های ارسال، ترافیک را از میان آن‌ها هدایت می‌کند.

²⁰forwarding device



شکل ۲-۵: ساختار NNFV مبتنی بر SDN [۲۶]

۲-۵ مسیریابی

مسیریابی عمل انتقال اطلاعات در میان یک شبکه، از یک منبع به مقصد است. در طول مسیر، حداقل یک گره میانی معمولاً قرار دارد. از این عمل همچنین به عنوان فرایند انتخاب مسیر برای ارسال بسته‌ها نیز یاد می‌شود. الگوریتم مسیریابی بخشی از نرم‌افزار لایه شبکه^{۲۱} است که مسئول تصمیم‌گیری در مورد این است که یک بسته در حال دریافت باید به کدام خط خروجی ارسال شود، یعنی گره بعدی برای بسته چه باید باشد.

پروتکل‌های مسیریابی از متریک‌هایی استفاده می‌کنند تا بتوانند بهترین مسیر را برای جابجایی یک بسته انتخاب کنند. متریک یک استاندارد اندازه‌گیری است؛ مانند پهنای باند مسیر، قابلیت اطمینان، تأخیر، بار فعلی روی آن مسیر و غیره؛ که توسط الگوریتم‌های مسیریابی، برای تعیین مسیر بهینه به مقصد، مورد استفاده قرار می‌گیرند. برای کمک به فرایند تعیین مسیر، الگوریتم‌های مسیریابی، جداول مسیریابی را که شامل اطلاعات مسیر هستند، تهیه و نگهداری می‌کنند. اطلاعات مسیر بسته به الگوریتم مسیریابی مورد استفاده متفاوت است. الگوریتم‌های مسیریابی جداول مسیریابی با انواع اطلاعات را پر می‌کنند. به طور عمده ترکیب‌های مقصد/هاپ

²¹network layer

بعدی، به روتر می‌گویند که می‌توان به‌طور مطلوب با ارسال بسته به یک گره خاص که نشان‌دهنده هاپ بعدی^{۲۲} در راه به مقصد نهایی است، به یک مقصد خاص رسید. هنگامی که روتر یک بسته ورودی را دریافت می‌کند، آدرس مقصد را بررسی می‌کند و تلاش می‌کند که این آدرس را با یک هاپ بعدی مرتبط کند. روترها، متریک‌ها را برای تعیین مسیرهای مطلوب مقایسه می‌کنند. این متریک‌ها بسته به طراحی الگوریتم مسیریابی مورد استفاده، متفاوت هستند. روترها با یکدیگر ارتباط برقرار می‌کنند و از طریق انتقال انواع پیام‌ها از جداول مسیریابی خود نگهداری می‌کنند.

پیام به‌روزرسانی مسیریابی، یک پیام است که عموماً از تمام یا بخشی از یک جدول مسیریابی تشکیل شده است. با تجزیه و تحلیل به‌روزرسانی‌های مسیریابی از همه روترهای دیگر، یک روتر می‌تواند تصویر دقیقی از توپولوژی شبکه ایجاد کند. پیام اعلان وضعیت لینک، مثال دیگری از یک پیام فرستاده‌شده بین روترها است. روترهای دیگر را از وضعیت لینک‌های فرستنده مطلع می‌کند. اطلاعات لینک همچنین می‌تواند برای ساخت یک تصویر کامل از توپولوژی شبکه استفاده شود تا روترها بتوانند مسیرهای بهینه به مقاصد شبکه را تعیین کنند.

متریک‌های الگوریتم مسیریابی

جداول مسیریابی شامل اطلاعاتی است که توسط نرم‌افزار سوئیچینگ برای انتخاب بهترین مسیر استفاده می‌شود. در این قسمت درباره ماهیت متفاوت اطلاعاتی که در آن جداول وجود دارند و نحوه‌ای که تعیین می‌کنند یک مسیر بر بقیه ترجیح دارد صحبت خواهد شد.

الگوریتم‌های مسیریابی برای تعیین بهترین مسیر از معیارهای مختلف استفاده می‌کنند. الگوریتم‌های پیچیده الگوریتم مسیریابی می‌توانند انتخاب مسیر را بر پایه چندین معیار، با ترکیب آن‌ها را در یک معیار تکی یا ترکیبی، انجام دهند. تمام معیارهای زیر می‌توانند استفاده شوند:

طول مسیر: طول مسیر معمول‌ترین متریک مسیریابی است. برخی از پروتکل‌های مسیریابی اجازه می‌دهد تا مدیران شبکه هزینه‌های دلخواه به هر لینک شبکه اختصاص دهند. در این مورد، طول مسیر مجموع هزینه‌هایی است که مربوط به هر لینکی است که عبور از آن رخ می‌دهد.

تعداد هاپ: دیگر پروتکل‌های مسیریابی تعداد هاب را تعریف می‌کنند، یک متریک که تعداد گذر یک بسته در یک مسیر از منبع به مقصد، از میان محصولات متصل به اینترنت مانند روترها، را مشخص می‌کند.

تأخیر مسیریابی: تأخیر مسیریابی به طول زمان لازم برای انتقال بسته از منبع به مقصد از طریق اینترنت اشاره دارد. تأخیر به بسیاری از عوامل، از جمله پهنای باند لینک‌های شبکه‌های میانی، صف‌های پورت (صف‌های

²²next hop

دریافت و انتقال موجود در روترها) در هر روتر در طول مسیر، تراکم شبکه در تمام لینک‌های میانی و مسافت فیزیکی که باید پیموده بستگی دارد. از آنجاکه تأخیر ترکیبی از چندین متغیر مهم است، یک متریک معمول و مفید است.

پهنای باند: پهنای باند به ظرفیت ترافیکی موجود یک لینک اشاره دارد. اگر همه چیزهای دیگر برابر باشد، یک لینک ۱۰ مگابیتی اترنت، به ترتیب به یک خط ۶۴ کیلوبیتی بر ثانیه ترجیح داده می‌شود. اگرچه پهنای باند یک شاخص از حداکثر گذردهی قابل‌دستیابی در یک لینک است، مسیرها از طریق لینک‌های با پهنای باند بیشتر لزوماً مسیرهای بهتری از مسیرهای از طریق لینک‌های کندتر ارائه نمی‌دهند. به عنوان مثال، اگر یک لینک سریع‌تر، شلوغ‌تر باشد، زمان واقعی موردنیاز برای ارسال یک بسته به مقصد می‌تواند بیشتر باشد.

بار: بار به میزانی که یک منبع شبکه، مانند یک روتر، مشغول است اشاره می‌کند. بار را می‌توان به روش‌های مختلفی، از جمله حد استفاده از CPU و بسته‌های پردازش‌شده در هر ثانیه محاسبه کرد. نظارت بر این پارامترها به طور مداوم، خود می‌تواند منابع را شدیداً درگیر کند.

هزینه: هزینه ارتباطات یکی دیگر از شاخص‌های مهم است، به خصوص به این دلیل که برخی از شرکت‌ها ممکن است در مورد عملکرد به اندازه هزینه‌های عملیاتی خود اهمیتی ندهند. مثلاً بسته‌ها را بیشتر از خطوط خود ارسال کنند و نه از طریق خطوط عمومی که برای زمان استفاده پول هزینه کنند حتی اگر تأخیر در خط خودشان بیشتر باشد.

قابلیت اطمینان: قابلیت اطمینان در زمینه الگوریتم‌های مسیریابی، به قابلیت اطمینان (معمولاً با توجه به میزان خطای بیت) هر لینک شبکه اشاره می‌کند. برخی از لینک‌های شبکه ممکن است بیشتر از سایرین دچار خطا شوند. پس از اینکه یک شبکه خراب می‌شود، برخی از لینک‌های شبکه ممکن است راحت‌تر یا سریع‌تر از سایر لینک‌ها تعمیر شوند. هر فاکتور اطمینانی می‌تواند در انتساب درجه‌بندی قابلیت اعتماد مورد استفاده قرار گیرد. این فاکتورها مقادیر عددی دلخواه هستند و معمولاً توسط مدیران شبکه به لینک‌های شبکه اختصاص داده می‌شوند. [۲۴]

۲-۵-۱ SDN در مسیریابی

مسیریابی ستون اصلی پروتکل TCP/IP است. به کمک الگوریتم‌های یافتن کوتاه‌ترین مسیر در یک گراف، شبکه کوتاه‌ترین مسیر به مقصد را پیدا می‌کند و ترافیک را به آن مقصدها می‌رساند. با این حال، در سیستم مسیریابی

فعلی، مشکلات زیادی وجود دارد. برای مثال:

- از آنجایی که شرایط بسیار سریع تغییر می‌کند، کوتاه‌ترین مسیری که پروتکل مسیریابی پیدا کرده است ممکن است کوتاه‌ترین مسیر دیگر نباشد.
- یافتن مسیر مناسب خود مستلزم صرف زمان است که این امر منجر به طولانی‌تر شدن زمان همگرایی می‌شود.
- پرش مسیر از دیگر مشکلات شایع است. پرش مسیر هنگامی رخ می‌دهد که روتر متناوباً یک شبکه مقصد را از طریق یک مسیر و سپس به سرعت از طریق مسیری دیگر تبلیغ می‌کند. پرش مسیر هنگامی اتفاق می‌افتد که شرایط آسیب‌زایی مانند خطای سخت‌افزار، نرم‌افزار یا پیکربندی در شبکه پیدا شود و موجب شود که اطلاعاتی خاص مکرراً اطلاع داده‌شده و از بین برود. مثلاً وجود پیشوندهای ناپایدار و بی‌ثبات از جمله عواملی هستند که می‌توانند موجب نوسانات مسیر شوند.
- درگیری‌ها و تضادهای اعمال سیاست مسیریابی درون سیستم‌های مستقل می‌تواند منجر به بی‌ثباتی یا حتی واگرایی سراسری شود.
- پیاده‌سازی سیاست‌های مسیریابی سنگین و پیچیده است. سیاست‌های سیستم‌های مستقل باید به گونه‌ای اظهار شوند تا کاملاً با مکانیسم‌های مسیریابی هماهنگ شوند، در صورتی که در حالت ایده‌آل عکس این حالت باید رخ دهد.
- مکانیسم‌های سیاست داخلی (IGP) و بیرونی (BGP) بر پایه یک طرح واحد ساخته و کنترل نمی‌شوند که این امر منجر به مشکلات بیشتر می‌شود.
- اپراتورها مجبور هستند مکانیسم‌های مسیریابی قرار داده‌شده را نیز سر بار کنند تا بتوانند سیاست‌های مختلف را پیاده‌سازی کنند.

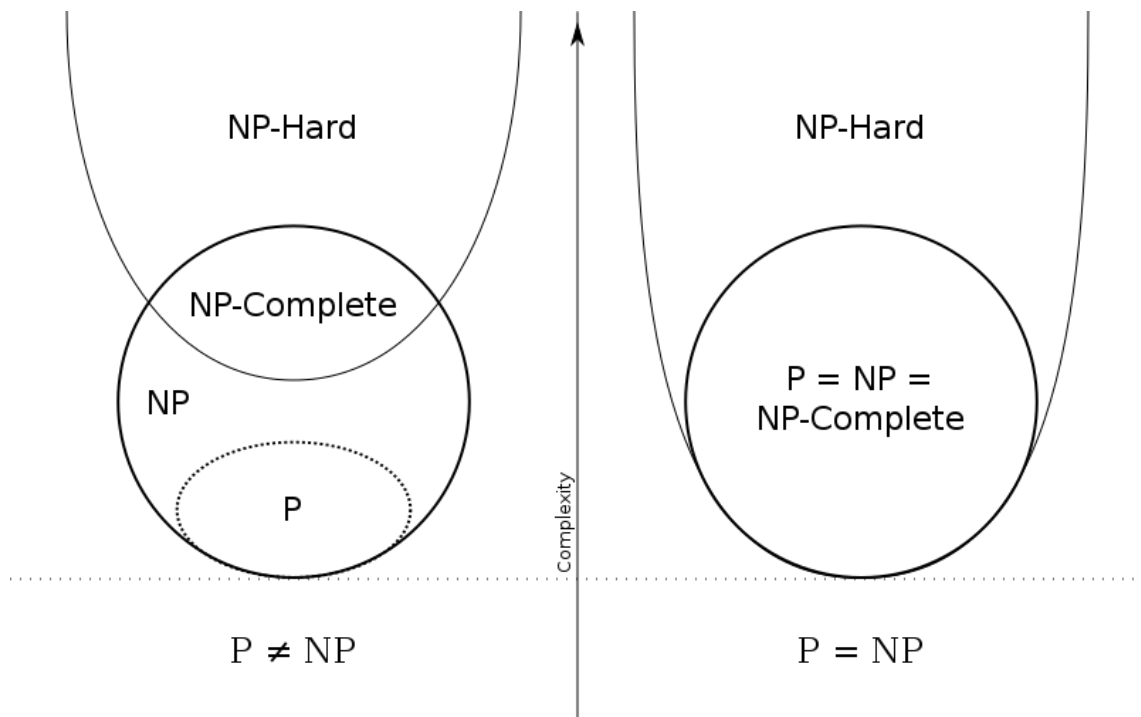
ایده SDN برای حل این مشکلات در معماری پروتکل TCP/IP ساده است: از آن‌جا که نمی‌توان یک شبکه بزرگ را با معماری توزیع‌شده کنونی مدیریت کرد، مدیریت متمرکز بهتر خواهد بود. SDN فرصت‌های جدیدی را ارائه می‌دهد. مفهوم کلیدی SDN، جداسازی لایه کنترل شبکه از لایه داده است. SDN یک سیستم عامل شبکه (NOS) را فعال می‌کند که با المان‌های ارسال بسته ارتباط برقرار می‌کند. ویژگی‌های کنترل و برنامه‌های

کاربرد، از جمله الگوریتم‌های مسیریابی، می‌توانند در بالای NOS مستقر شوند و به عنوان ماژول‌های نرم‌افزاری اجرا شوند. NOS نمایی پایدار از کل شبکه را به منطق کنترلی متمرکزی که روی آن اجرا می‌شود ارائه می‌دهد. جداسازی کنترل شبکه از لایه داده و نتیجتاً تمرکز منطقی لایه کنترل مسیریابی، می‌تواند به طور قابل توجهی مدیریت مسیریابی را در یک سیستم مستقل ساده‌تر کند و همگرایی سریع‌تر مسیریابی را فراهم کند. مفهوم اصلی این است که کنترل تصمیمات داخلی و خارجی مسیریابی مرتبط با یک سیستم مستقل متمرکز شود. با انجام این کار، مزایای متعددی به دست می‌آید.

- می‌توان مسیریابی مبتنی بر سیاست ساده‌تری پیاده‌سازی کرد، زیرا بیان، اعمال و بررسی سیاست‌های مسیریابی به صورت مرکزی ممکن خواهد بود.
 - تمرکز منطقی از طریق کاهش پیچیدگی کلی مدیریت، به بهبود مقیاس‌پذیری مسیریابی در یک سیستم مستقل کمک می‌کند.
 - یک کنترل‌کننده SDN مرکزی، اصلاح پروتکل‌های مسیریابی را ساده می‌کند، زیرا این کار می‌تواند صرفاً از طریق نرم‌افزار صورت گیرد.
 - می‌توان یک معماری مسیریابی در حال تکامل داشت، زیرا پروتکل‌های مسیریابی درون دامنه‌ای می‌توانند خیلی راحت‌تر تغییر کنند.
 - شبیه‌سازی لایه کنترل جهت پیاده‌سازی ایمن تغییرات پیکربندی ساده‌تر می‌شود.
 - برای تعادل بار و اهداف پیشگیرانه، می‌توان کنترل‌کننده‌های مسیریابی چندگانه‌ای روی همان شبکه داشت.
- پس می‌توان گفت که مسیریابی مرکزی شده شرایط مطلوب‌تری جهت پیاده‌سازی کارهای اینترنتی و بهینه‌سازی شبکه فراهم می‌آورد. [۳۸]

۶-۲ روش‌های اکتشافی

مسائل بهینه‌سازی را می‌توان با توجه به زحمت محاسباتی‌ای که برای حل آن‌ها لازم است دسته‌بندی کرد. مسائلی که در دسته P هستند معمولاً آسان‌اند، زیرا الگوریتم‌های شناخته‌شده‌ای وجود دارند که این مسائل را در زمان



شکل ۲-۶: نمودار اولر برای کلاس‌های مختلف پیچیدگی [۳]

چندجمله‌ای حل می‌کنند. مسائل دسته NP مسائلی اند که می‌توانند توسط یک الگوریتم نامعین، در بدترین حالت، در زمان چندجمله‌ای حل شوند. مسائلی که در دسته NP کامل اند، پیچیده هستند، چون هیچ الگوریتمی با زمان چندجمله‌ای برای آن‌ها شناخته شده نیست. مسائل تصمیم‌گیری‌ای که در NP نیستند، یعنی مسائل دسته NP دشوار، حتی سخت‌تر نیز هستند، زیرا در زمان چندجمله‌ای نمی‌توان ارزیابی کرد که آیا یک راه‌حل خاص برای چنین مسائلی میسر است یا خیر [۳۶]. در شکل ۲-۶ می‌توان رابطه این کلاس‌های مختلف را با یکدیگر مشاهده نمود.

نتایج مربوط به NP دشوار بودن یک مساله در علوم کامپیوتری نظری، روش‌های اکتشافی را تبدیل به تنها گزینه قابل قبول برای انواع مسائل بهینه‌سازی پیچیده که باید مرتباً در برنامه‌های دنیای واقعی حل شوند، می‌سازد. در علوم کامپیوتر، هوش مصنوعی و بهینه‌سازی، الگوریتم‌های اکتشافی روشی برای حل مسائلی اند که راه‌های کلاسیک حل آن‌ها بسیار کند می‌باشند و یا به عنوان راه‌حل تقریبی برای مسائلی به کار می‌روند که راه‌های کلاسیک نمی‌توانند برای آن‌ها جواب دقیقی پیدا کنند. هدف یک الگوریتم اکتشافی این است که یک راه‌حل که به اندازه کافی برای حل مساله مناسب است را در یک زمان معقول تولید کند. این راه‌حل ممکن است بهترین راه‌حل برای

مساله مذکور نباشد و یا فقط تخمینی از راه حل دقیق باشد اما هنوز ارزشمند است، زیرا یافتن آن نیازی به زمان طولانی ندارد. الگوریتم های اکتشافی ممکن است مستقیماً نتایج خود را تولید کنند و یا ممکن است در ترکیب با سایر الگوریتم های بهینه سازی برای بهبود کارایی آن ها استفاده شوند [۲].

معیارهایی که هنگام تصمیم گیری برای استفاده از یک روش اکتشافی برای حل یک مساله مشخص در نظر گرفته می شوند، شامل موارد زیر می باشند.

- تضمین کند که به یک پاسخ برای مساله دست پیدا می کند.
 - در یک زمان منطقی پاسخ را پیدا کند، به این معنا که پاسخ از نظر محاسباتی قابل دست یابی باشد.
 - عموم داده ها را به عنوان ورودی بپذیرد.
 - بین پاسخ و مساله پیوستگی ایجاد کند.
- نکات فوق استدلال محکمی در مورد چرایی انتخاب هر روش اکتشافی ارائه می دهند، زیرا هر رویکردی که نتواند شرایط فوق را برآورده سازد، به طور جامع قادر به برآورده کردن نیازهای کاربر نیست.
- بدیهی است که عدم برقراری موارد اول و دوم قابل قبول نخواهد بود. شرط سوم نیز از اهمیت زیادی برخوردار است، زیرا اگر پاسخی شرط سوم را برقرار نسازد کلیت و انعطاف پذیری خود را از دست می دهد. همچنین عدم برقراری شرط آخر، بازیابی داده ها و آزمایش راه حل های جایگزین در طی فرایند حل مساله را برای دیگر کاربران مشکل می کند.

۷-۲ پیشنهاد پژوهش

پژوهش [۷]، نشان دهنده یک چشم انداز شامل چند ذینفع از جمله کاربر، ارائه دهنده خدمات و ارائه دهنده زیرساخت^{۲۳} است. بنابراین مساله جایابی VNF ها برای به حداکثر رساندن تعداد درخواست های پذیرفته شده از مجموعه درخواست های دریافتی و همچنین، به حداکثر رساندن رضایت مندی مشتریان در نظر گرفته می شود. مدل ارائه شده در این پژوهش، همچنین درخواست های خدمات را با سطوح اولویت بندی متمایز کرده و تضمین می کند که اهداف کیفیت خدمات برای درخواست های پذیرفته شده خدمات برآورده می شوند.

²³Infrastructure provider

روش جاسپر در [۱۰]، یک رویکرد خودکار برای مقیاس‌دهی، جایابی و مسیریابی همزمان سرویس‌های شبکه است که هدف آن حداقل کردن موارد نقض قیود، مثل CPU حافظه و محدودیت‌های ظرفیت لینک است که با در نظر گرفتن مجموعه‌ای از اهداف ثانویه، همچون تأخیر کل، مصرف منابع و غیره، یک بهینه‌گی پارتو است. در [۲۳] جایابی بهینه زنجیره‌های VNF بررسی شده و نشان داده می‌شود که مساله می‌تواند برای موارد بسیار خاص NP کامل باشد.

در [۲۱]، مساله بهینه‌سازی جایابی پویا توابع شبکه و مسیریابی جریان در یک زنجیره از توابع شبکه در نظر گرفته می‌شود. در این پژوهش، برای به حداکثر رساندن نرخ جریان قابل قبول و به حداقل رساندن هزینه انرژی برای چندین زنجیره سرویس، یک مساله بهینه‌سازی چند تابع هدفه تدوین می‌شود. مساله بهینه‌سازی چند تابع هدفه به یک مساله برنامه‌نویسی خطی عدد صحیح مختلط (MILP) تبدیل می‌شود و ثابت می‌شود که مساله NP-hard است.

[۱۷] برای به حداقل رساندن تعداد قوانین مسیریابی برای زمانی که تمام توابع سرویس یک زنجیره درون یک ماشین مجازی واحد قرار گرفته‌اند، مساله بهینه‌سازی مسیر را فرمول‌بندی می‌کند. این کار ترتیب پردازش مشخص شده در زنجیره‌های سرویس را در نظر نمی‌گیرد.

[۲۹] مساله جا دادن جریان‌های بیشتر در یک دامنه، با به حداقل رساندن حداکثر استفاده از لینک‌ها و CPU ها را فرمول‌دهی می‌کند و چندین روش اکتشافی پیشنهاد می‌کند. این رویکرد به جای حداکثر رساندن نرخ جریان قابل قبول، به حداکثر رساندن ظرفیت منابع باقی‌مانده در هر پیوند و CPU را هدف قرار می‌دهد.

برخی پژوهش‌ها، مانند [۳۹]، هم جایابی و هم مسیریابی جریان را در نظر می‌گیرند، اما آن‌ها را جداگانه حل می‌کنند. به عنوان مثال، یک روش اکتشافی برای جایابی استفاده می‌شود و از نتیجه آن به عنوان ورودی برای هدایت جریان استفاده می‌شود.

[۳۲] از یک فرمول‌بندی آنالین و آفلاین استفاده می‌کند. تمرکز اصلی فرمول‌بندی آفلاین محدود کردن میزان قوانین ارسال به دلیل محدودیت در حافظه TCAM سوئیچ‌های SDN است. فرمول‌بندی آنالین برای تعادل بار آنالین بر روی سوئیچ‌های موجود است. این پژوهش، امکان راه‌اندازی پویا سرویس‌ها را بررسی نمی‌کند.

[۱۶] جایابی و مسیریابی زنجیره توابع مجازی شبکه را با تمرکز بر به حداقل رساندن ظرفیت لینک استفاده شده شبکه بررسی می‌کند.

در پژوهش [۴]، نویسندگان یک مدل شبکه NFV مناسب برای عملیات ISP^{۲۴} ارائه می‌دهند. آن‌ها مساله

²⁴Internet Service Provider

بهینه‌سازی مسیریابی زنجیره VNF را تعریف می‌کنند و یک فرمول‌بندی برنامه ریزی خطی عدد صحیح مخلوط را ارائه می‌کنند.

[۳۳] یک چارچوب بهینه‌سازی مسیریابی، با آگاهی از قابلیت اطمینان و دارای تاخیر محدود به نام (READ) را برای شبکه‌های داده دارای NFV ایجاد می‌کند. READ شامل فرمول‌بندی یک برنامه‌نویسی خطی مختلط عدد صحیح (MILP) است که منجر به جایابی VNF و مسیریابی ترافیک می‌شود و به طور همزمان، قابلیت اطمینان حاصل از خدمات پشتیبانی شده توسط شبکه را به حداکثر می‌رساند.

در [۱۴]، مساله جایابی توابع شبکه مجازی و مسیریابی در میزبان‌های فیزیکی مورد مطالعه قرار می‌گیرد تا تأخیر کلی تعریف شده به صورت تاخیر صف در لینک‌های شبکه را به حداقل برساند. از این نظر، این پژوهش هم مساله زنجیره کردن و جایابی VNF ها و هم جنبه مسیریابی جریان را در نظر می‌گیرد.

در [۲۷]، تأثیرات تغییر در ترافیک جعبه‌های میانی مورد مطالعه قرار می‌گیرد و راه‌حلی برای جایابی جعبه‌های میانی مبتنی بر SDN جهت دستیابی به تعادل بار مطلوب پیشنهاد می‌شود. مساله جایابی VNF ها با آگاهی از ترافیک (TAMP) به عنوان یک مساله بهینه‌سازی با هدف به حداقل رساندن حداکثر نسبت بار لینک فرمول‌بندی می‌شود.

فصل ۳

هماهنگ سازی، مسیریابی و جایابی VNF

۱-۳ مقدمه

در این بخش ابتدا یک ساختار سلسله مراتبی برای سیستم ارائه می شود که تعامل بین بخش های مختلف را توضیح می دهد. سپس مفروضات شبکه مطرح می شود و مساله بهینه سازی فرمول بندی می شود و در ادامه توابع هدف و قیود مختلف مورد بررسی قرار می گیرند.

۲-۳ ساختار کلی سیستم

هدف اصلی این بخش اشاره به بلوک های کاربردی مورد نیاز و ارائه بستری جهت اجرای الگوریتم های جایابی VNF است. بنابراین، هدف اصلی این بخش اشاره به بلوک های کاربردی مورد نیاز و ارائه بستری جهت اجرای الگوریتم های جایابی VNF است. [۲۵]

پنج بخش اصلی در ساختار معرفی شده اند که در طول جایابی و راه اندازی VNF ها با یکدیگر تعامل دارند. شکل ۱-۳ این ساختار را نشان می دهد. این بخش ها عبارتند از: کاتالوگ خدمات، هماهنگ ساز NFV، بهینه ساز منبع، کنترل کننده SDN و کنترل کننده ابری.

- کاتالوگ خدمات: کاربران به منظور انتخاب مجموعه توابعی که نیازهایشان را برآورده سازند، به کاتالوگ

خدمات دسترسی پیدا می کنند. نمونه هایی از خدمات ارائه شده توسط کاتالوگ می تواند شامل عناصر سوئیچینگ، فایروال ها و امنیت وب، سیستم های تشخیص و جلوگیری از نفوذ،^۱ توابع موجود در روترهای خانگی، توابع آنالیز ترافیک، امنیت و توابع توزیع کلید در بین دیگران باشد.

- هماهنگ ساز NFV: همانطور که توسط گروه مشخصات صنعتی NFV از موسسه ETSI نیز مشخص شده است، هسته اصلی معماری، هماهنگ سازی NFV است. این جز، با ساختاری خاص توسط هماهنگ ساز NFV به تصویر کشیده شده است و وظیفه مدیریت کلی چرخه عمر VNF ها را بر عهده دارد. هماهنگ ساز NFV از طریق رابط شمالی^۲ با کاتالوگ خدمات ارتباط برقرار می کند. از طریق این رابط، هماهنگ ساز، کاتالوگ خدمات را با NF های ارائه شده به روز می کند و همچنین اطلاعات مربوط به خدمات انتخاب شده از مشتریان را بازیابی می کند.

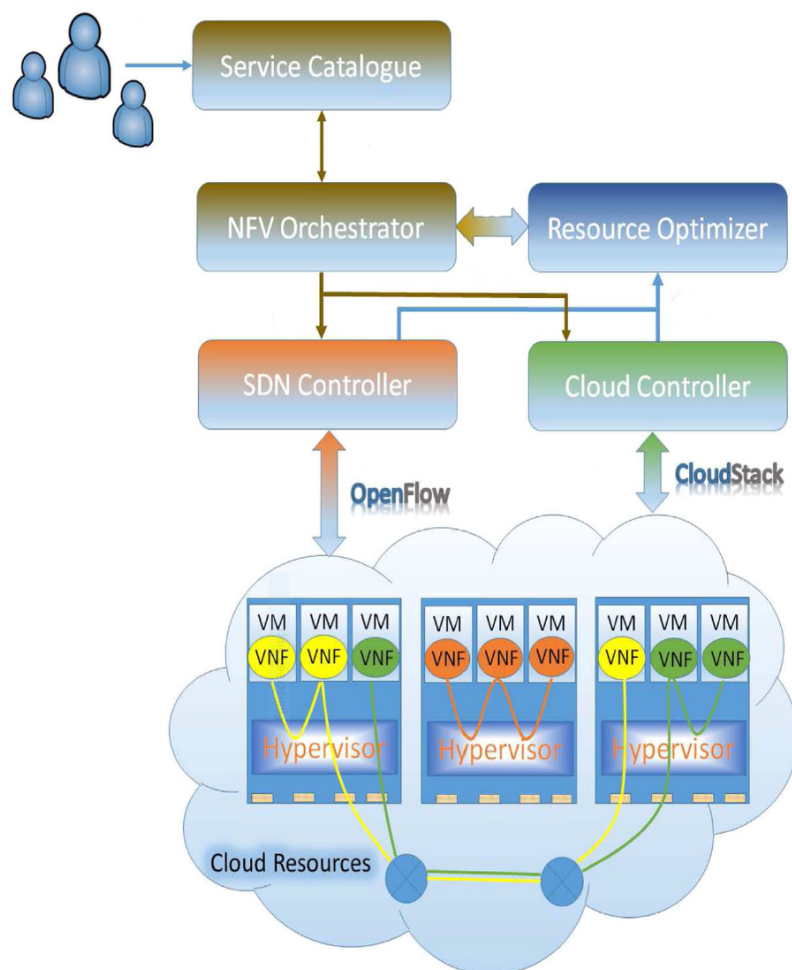
- بهینه ساز منبع: جهت تضمین پیکربندی، جایابی و زنجیره کردن VNF ها، هماهنگ ساز نیاز به برقراری ارتباط با جز بهینه ساز منابع دارد. این مؤلفه وظیفه دارد بر اساس خدماتی که باید مستقر شوند، مکان های بهینه سرور را در زیرساخت ابری انتخاب کند. کار اصلی بهینه ساز منبع، اجرای الگوریتم های مناسب برای تخصیص VNF های انتخاب شده است. برای تأمین چنین جایابی بهینه ای از VNF ها در زنجیره خدمات، بهینه ساز منبع به اطلاعات به روز وضعیت منابع، از جمله ظرفیت موجود منابع ابری و تقاضاهای VNF ها به عنوان اطلاعات ورودی نیاز دارد. در ادامه، راه حل مساله جایابی VNF به هماهنگ ساز NFV بازگردانده می شود.

- کنترل کننده SDN: کنترلر SDN برای ایفای نقش یک ناظر متمرکز در شبکه تعبیه شده است. این بخش دارای چشم اندازی از کل شبکه است و دستگاه های ارسال قرار گرفته در زیرساخت ابری را به وسیله اجرا و نگهداری محاسبات مسیریابی به منظور اجرای زنجیره خدمات بین VNF ها کنترل می کند. بنابراین کنترلر SDN وظیفه ارائه وضعیت منابع شبکه به بهینه ساز منبع و همچنین به روزرسانی و تشکیل جداول ارسال دستگاه های سوئیچینگ با توجه به نتیجه جایابی VNF که توسط هماهنگ ساز فراهم می شود را بر عهده دارد. OpenFlow پروتکل استاندارد است که برای پیکربندی جدول های جریان منابع شبکه زیرساخت مورد استفاده قرار می گیرد.

¹Intrusion Prevention Systems

^۲interface northbound

- کنترل کننده ابری: مسئول پیکربندی منابع محاسباتی زیرساخت ابری است. این بخش، وضعیت و عملکرد سرورهای موجود را حفظ و بازیابی می کند تا ظرفیت فیزیکی موجود سرورها (به عنوان مثال CPU، حافظه و غیره) را به بهینه ساز منبع ارائه دهد. همچنین، این واحد نتیجه جایابی VNF را از هماهنگ ساز NFV دریافت می کند تا سرورهای مناسب را برای VNF ها فراهم کند.



شکل ۳-۱: ساختار کلی سیستم [۲۵]

ساختار معرفی شده، با چارچوب ساختار معرفی شده توسط ETSI منطبق است. بطور کلی سه حوزه فعالیت اصلی مشخص شده توسط ETSI وجود دارد: توابع مجازی شبکه (VNF)، زیرساخت NFV (NFVI) و مدیریت و هماهنگ سازی NFV (MANO) که می توانند در ساختار معرفی شده نیز مشخص شوند. ساختار سیستم معرفی شده سعی بر این دارد که بیشتر به توصیف ویژگی های اصلی که باید در حین استاندارد

سازی NFV مورد توجه قرار گیرند بپردازد. در این حالت، هدف نهایی تضمین قابلیت همکاری بین فن آوری های مختلف درگیر، همچون SDN، ضمن معرفی مزایای متعدد NFV (به عنوان مثال جداسازی نرم افزار از سخت افزار، به کارگیری انعطاف پذیر سرویس و غیره) است.

به طور خاص، بلوک های اصلی عملکردی معرفی شده توسط ETSI به این صورت تبدیل می شوند: VNF ها یکسان باقی می مانند و به مجازی سازی توابع یا ویژگی های خدمات که در داخل VM ها ساکن هستند اشاره دارند. ساختار NFV به کل سخت افزارها و منابع مجازی اشاره دارد که به عنوان زیرساخت منابع ابری در شکل ۱-۳ نشان داده شده است. مدیریت و هماهنگ سازی NFV (MANO) هماهنگ سازی و مدیریت منابع فیزیکی موجود در زیرساخت و مدیریت VNF ها را پوشش می دهد که اکنون، در ساختار معرفی شده، این وظایف توسط هماهنگ ساز NFV، بهینه ساز منبع، کنترل کننده های SDN و ابری اجرا می شوند. شکل ۲-۳ نشان می دهد که چگونه مدل ساختار ETSI با چارچوب معماری معرفی شده مقایسه شده است.

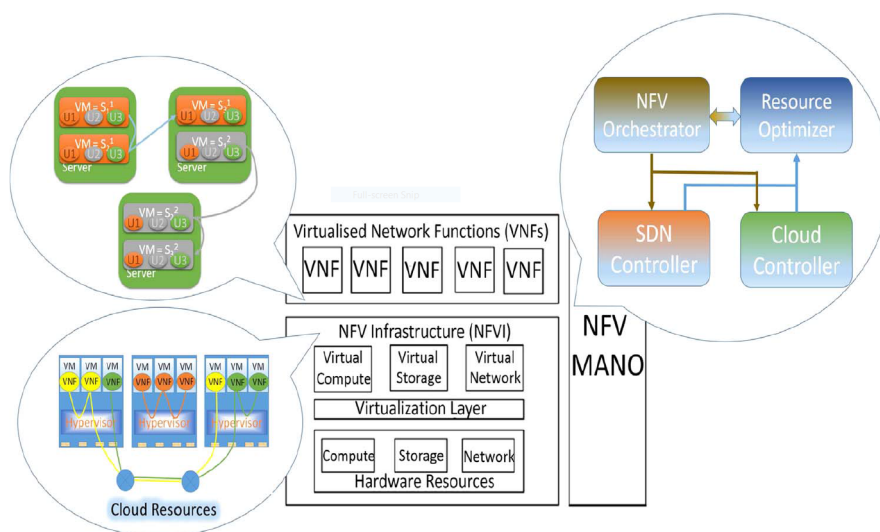
بنابراین برخلاف مدل های استاندارد سازی موجود، بلوک عملکردی بهینه ساز منبع معرفی شده است که به چالش های مدیریت استقرار^۳ VNF، از جمله یافتن منابع مناسب، تغییر مقیاس منابع سخت افزاری و پشتیبانی از یک مدل منبع پویا رسیدگی می کند. این امر می تواند از طریق قابلیت همکاری با SDN و کنترل کننده ابر حاصل شود که همزیستی هماهنگ همه فناوری های شبکه را تضمین می کند. در نهایت، سرانجام، توزیع عملکردها و نقش ها بین بخش های مختلف چارچوب معماری معرفی شده بسیار واضح تر است و می تواند مدیریت و هماهنگ سازی VNF ها را تسهیل کند.

۳-۳ بیان مساله

در این بخش روند جایابی VNF را توضیح می دهیم که توسط بهینه ساز منبع انجام می شود. زیرساخت منابع ابری به صورت یک گراف وزن دار بدون جهت مدل شده است و توسط $G = (V, I)$ نشان داده می شود که در آن V نمایانگر مجموعه گره ها و E مجموعه لینک های موجود است. [۲۵]

در این ساختار، گره ها متشکل از دو دسته کلی هستند: (۱) سرورها $N \subset V$ که برای میزبانی VNF به عنوان ماشین های مجازی استفاده می شوند و (۲) سوئیچ ها $X \subset V$ که زنجیره کردن VNF ها بین سرورهای پراکنده از نظر جغرافیایی را ممکن می سازند، از این رو $X \cup N = V$. براساس نوع، هر گره $v \in V$ با یک

³deployment



شکل ۳-۲: تطبیق ساختار سیستم با ETSI [۲۵]

بردار از ظرفیت‌های منابع موجود (مثل CPU، حافظه و انباره برای سرورها و های LAN مجازی و ورودی‌های جریان برای سویچ‌ها) نسبت داده می‌شود. در این جا، بردار شامل مشخصه CPU می‌باشد. همچنین، هر لینک $(u, v) \in E$ دارای یک ظرفیت پهنای باند موجود است که با $bw(u, v)$ نشان داده می‌شود.

قرار دادن VNF تخصیص VNF های یک زنجیره سرویس بر روی سرورهای مناسب زیرساخت ابری را نشان می‌دهد. به طور خاص، یک زنجیره سرویس را در نظر می‌گیریم که شامل k عدد VNF است که نیاز به راه اندازی دارند تا یک عملکرد خاص را شروع کنند.

علاوه بر این، هر VNF در یک زنجیره خدمات، ترافیک را به VNF بعدی در زنجیره منتقل می‌کند و باعث ایجاد ارتباط بین عناصر متوالی مجموعه زنجیره می‌شود که شامل خواسته های ورودی و خروجی بین VNF ها است.

هر VNF در زنجیره خدمات همچنین با یک بردار از مطالبات محاسباتی با نام D (مثل CPU، حافظه، انباره و غیره) مشخص می‌شود که باید برآورده شوند. اندازه مطالبات با نوع VNF متناسب است.

جهت جایابی VNF، یک تابع هدف مناسب باید تعیین شود. همانطور که قبلاً نیز ذکر شد، NFV به عنوان ابزاری برای کاهش هزینه‌های عملیاتی ارائه شده است. بنابراین این تابع هدف می‌تواند از نوع هزینه باشد. برای تعریف آن، می‌توان صرفاً منافع فراهم‌آورنده خدمات مخابراتی را تامین کرد و یا علاوه بر آن، منافع فراهم‌آورنده زیرساخت را نیز در نظر گرفت. فراهم‌آورنده زیرساخت یا InP، منابع فیزیکی را به صورت مراکز داده و شبکه‌های

فیزیکی پیاده سازی و مدیریت می کند. وجود این منابع فیزیکی ضروری است تا منابع مجازی فراهم شده و از طریق رابط های برنامه نویسی به یک یا چند TSP اجاره داده شوند. InP ها همچنین می توانند چگونگی تخصیص منابع موجود به TSP ها را تعیین کنند. در NFV نمونه هایی از InP ها می توانند مراکز داده ای عمومی مانند آمازون (Amazon) یا سرورهای خصوصی متعلق به TSP ها باشند. اگر یک InP مشخص، به طور کلی یا جزئی، قادر به ارائه منابع برای یک TSP مشخص نباشد، مذاکرات و در نتیجه ائتلاف هایی با سایر InP ها می توانند شکل گرفته شوند تا بتوانند VNF های چند دامنه را فراهم کنند.

فراهم آورنده خدمات مخابراتی منابع را از یک یا چند InP که از آن منابع برای راه اندازی VNF ها استفاده می کنند، اجاره می کنند. آن ها همچنین زنجیره ای از این توابع را برای ایجاد خدمات به کاربران نهایی^۴ تعیین می کنند. در یک حالت کلی تر، TSP ها ممکن است منابع مجازی خود را به سایر TSP ها اجاره دهند. در چنین مواردی، TSP فروشنده نقش یک InP را بازی می کند. در مواردی که InP خصوصی یا خانگی است، مثلاً حالتی که توسط گره های شبکه یا سرورهای TSP فراهم شده باشد، InP و TSP ممکن است یک نهاد باشند. جهت جایابی، VNF یک تابع هزینه مناسب باید تعیین شود که کارایی راه حل را هم از نظر استفاده از منابع ابری و هم از نظر هزینه ارائه دهنده خدمات نشان دهد.

برای ارائه دهندگان خدمات مطلوب است تا آن جا که ممکن است منابع کمتری را اجاره کند تا کاهش هزینه های عملیاتی را حداکثر نماید. از نقطه نظر ارائه دهنده خدمات، هدف نهایی حداقل کردن جمع هزینه های فعال سازی است و به شرح ۱-۳ می باشد. [۲۵]

$$\min \sum_{n \in N} R_n + \sum_{(u,v) \in V} R_{(u,v)} \quad (1-3)$$

هر سرور با هزینه فعال سازی سرور برای استقرار VNF ها همراه است که با $R_n, n \in N \subset V$ مشخص می شود. علاوه بر این، هنگامی که زنجیره سرویس میان چندین سرور وجود دارد، استفاده از لینک هایی که به سرورها وصل می شوند نیز با هزینه فعال سازی همراه است که با $R_{(u,v)}, (u,v) \in V$ مشخص می شوند. لازم به ذکر است که معادله ۱-۳، با تلاش برای به حداقل رساندن تعداد کل منابع مورد استفاده، صرفاً منفعت ارائه دهنده خدمات را در نظر می گیرد.

از سوی دیگر، صاحب زیرساخت های ابری باید با استفاده از یک روش متعادل کننده مناسب، از منابع

⁴end users

محاسباتی و شبکه‌ای موجود به صورت بهینه استفاده کند تا درآمد خود را در دراز مدت به حداکثر برساند. بهره‌برداری از این منابع باید به صورت بهینه صورت گیرد تا نیازهای ارائه‌دهنده ابر نیز تامین شود. با استفاده از ظرفیت‌های محاسباتی و شبکه‌ای کمتر بارگذاری شده از طریق نظارت بر ظرفیت موجود (مثلاً از طریق SDN و کنترل ابر) می‌توان یک طرح مناسب جهت متعادل‌سازی بار ارائه داد. بنابراین اگر $C(n)$ نشان‌دهنده ظرفیت سرور n و $bw(u, v)$ نشان‌دهنده پهنای باند لینک (u, v) باشد، دو ضریب برای معادله ۳-۱ معرفی می‌شوند که برابر با مقادیر معکوس ظرفیت موجود گره‌ها و پهنای باند لینک‌ها می‌باشند.

$$A_n = \frac{1}{C(n)}, B_{(u,v)} = \frac{1}{bw(u, v)} \quad (2-3)$$

با این روش، منابع کمتر استفاده شده برای جایابی VNF انتخاب می‌شوند که به ارائه‌دهنده ابر امکان استفاده بهینه از منابع موجود را می‌دهد. اکنون تابع هزینه می‌تواند به شرح ۳-۳ باشد. [۲۵]

$$\min \sum_{n \in N} A_n R_n + \sum_{(u,v) \in V} B_{(u,v)} R_{(u,v)} \quad (3-3)$$

۴-۳ فرمول بندی بهینه سازی خطی و مختلط عدد صحیح

فرمول‌بندی بهینه‌سازی خطی و مختلط عدد صحیح یک روش انعطاف‌پذیر و دقیق ریاضی برای فرمول‌بندی مسائل عمومی شبکه فراهم می‌کند. هرچند مسائل MILP به ویژه برای آزمایشات در مقیاس بزرگ، از نظر محاسباتی غیر قابل دنبال کردن هستند. بنابراین، راه‌حلی که در ادامه ارائه می‌شود برای آزمایشات با مقیاس کوچک‌تر، یعنی جایی که نیازهای محاسباتی کمتر است، مناسب‌تر است.

برای حل MILP، یک تابع هدف با چند قید را در نظر می‌گیریم. در این جا VNF های زنجیره سرویس را نیز به صورت گره‌های شبکه در نظر می‌گیریم و به این ترتیب گراف شبکه را کامل می‌کنیم. فرض بر این است که هر VNF از زنجیره S که با s_k نشان داده می‌شود با تمام سرورهای موجود $n \in N$ مرتبط است. بنابراین، گراف جدید تکمیل شده شبکه با $G' = (V', E')$ نشان داده می‌شود که در آن $V' = s_k \cup V$ و $E' = (s_k, n) \cup E$ ، $n \in N$ می‌باشد. هر اتصالی بین دو VNF پی در پی در زنجیره سرویس، به صورت جریانی که از VNF k ام شروع شده، گراف تکمیل شده G' را طی می‌کند و در نهایت به VNF $k + 1$ ام وارد می‌شود، تفسیر می‌شود. [۲۵]

متغیرهای مساله:

$h_{uv}^{s_k, s_{k+1}}$: یک متغیر باینری است که اگر جریان بین های VNF k ام و $k+1$ ام از طریق لینک $(u, v) \in \mathbf{E}'$ مسیردهی شود، یک خواهد بود و در غیر این صورت برابر با صفر خواهد بود.

$f_{uv}^{s_k, s_{k+1}}$: نشان دهنده میزان ترافیکی است که بین های VNF k ام و $k+1$ ام از طریق لینک $(u, v) \in \mathbf{E}'$ عبور می کند.

در تابع هدف معادله ۳-۴، هدف به حداقل رساندن هزینه فعال سازی لینک ها و سرورهایی است که برای جایابی زنجیره خدمات و مسیریابی جریان انتخاب شده اند و در آن هم منفعت فراهم آورنده خدمات و هم منفعت فراهم آورنده زیرساخت در نظر گرفته شده است. وزن های \mathbf{A}_n و $\mathbf{B}_{(uv)}$ نیز برای تضمین تعادل بار بین منابع موجود درج شده اند. در صورتی که فقط منفعت فراهم آورنده خدمات در نظر گرفته شود، این ضرایب حذف می شوند.

تابع هدف

$$\sum_{k=0 \dots K-1} \sum_{n \in N} \sum_{s_p \in S} \mathbf{A}_n \cdot \mathbf{R}_n \cdot h_{s_p n}^{s_k, s_{k+1}} + \sum_{k=0 \dots K-1} \sum_{(u,v) \in \mathbf{E}} \mathbf{B}_{(uv)} \cdot \mathbf{R}_{(uv)} \cdot h_{uv}^{s_k, s_{k+1}} \quad (۳-۴)$$

قیود

- قید ۳-۵ اطمینان حاصل می کند که مجموع ظرفیت های درخواست شده از جانب VNF ها که با $D(s_p)$ مشخص شده است، از ظرفیت موجود در سرورهای انتخاب شده که با $C(n)$ مشخص شده، تجاوز نمی کنند.

$$\sum_{s_p \in S} \mathbf{D}(s_p) \cdot h_{s_p n}^{s_k, s_{k+1}} \leq \mathbf{C}(n) \quad \forall n \in N, s_k, s_{k+1} \in S \quad (۳-۵)$$

- در قید ۳-۹، تضمین می شود که حداکثر ازدحام روی هر لینک شبکه، از یک مقدار مشخص بیشتر نشود. صف تشکیل شده برای هر لینک از نوع $M/M/1$ در نظر گرفته می شود. در صف $M/M/1$ ، ازدحام از رابطه ۳-۶ به دست می آید که در آن λ نشان دهنده نرخ ورود به صف و μ نشان دهنده نرخ پردازش می باشد.

$$\frac{\lambda}{\mu - \lambda} \quad (۳-۶)$$

اگر حداکثر ازدحام قابل تحمل روی هر لینک را برابر با اندازه بافر آن لینک در نظر بگیریم و با $bfr(u, v)$ نشان دهیم، در این صورت باید برای هر لینک رابطه ۷-۳ برقرار باشد.

$$\frac{\lambda}{\mu - \lambda} < bfr(u, v) \quad (7-3)$$

با مقایسه رابطه ۷-۳ با فرمول بندی مساله حاضر، نتیجه می شود که پهنای باند لینک (u, v) ، که با $bw(u, v)$ نشان داده می شود، معادل نرخ پردازشی آن لینک می باشد. همچنین نرخ ورود برای هر لینک (u, v) از رابطه ۸-۳ به دست می آید.

$$\sum_{k=0 \dots K-1} (f_{uv}^{s_k, s_{k+1}} + f_{vu}^{s_k, s_{k+1}}) \quad (8-3)$$

با قراردادی معادل های مساله به جای مقادیر μ و λ در رابطه ۷-۳، در نهایت قید حداکثر ازدحام روی هر لینک به شکل رابطه ۹-۳ خواهد بود.

$$\sum_{k=0 \dots K-1} (f_{uv}^{s_k, s_{k+1}} + f_{vu}^{s_k, s_{k+1}}) \leq \frac{bfr(u, v)}{1 + bfr(u, v)} bw(u, v), \quad \forall (u, v) \in \mathbf{E}' \quad (9-3)$$

- حفاظت از جریان از طریق مجموعه قیود ۱۰-۳، ۱۱-۳، ۱۲-۳ ارضا می شود. این قیود، به ترتیب، نشان دهنده مجموع جریان های ورودی و خروجی برای نودهای میانی، مقصد و مبدا می باشند.

$$\sum_{w \in \mathbf{V}'} f_{uw}^{s_k, s_{k+1}} - \sum_{w \in \mathbf{V}'} f_{wu}^{s_k, s_{k+1}} = 0, \quad \forall k \in (0 \dots K-1), \quad \forall u \in \mathbf{V}' \setminus s_k, s_{k+1} \quad (10-3)$$

$$\sum_{w \in \mathbf{V}'} f_{s_k w}^{s_k, s_{k+1}} - \sum_{w \in \mathbf{V}'} f_{w s_k}^{s_k, s_{k+1}} = bw(s_k, s_{k+1}), \quad \forall k \in (0 \dots K-1) \quad (11-3)$$

$$\sum_{w \in \mathbf{V}'} f_{s_{k+1} w}^{s_k, s_{k+1}} - \sum_{w \in \mathbf{V}'} f_{w s_{k+1}}^{s_k, s_{k+1}} = -bw(s_k, s_{k+1}), \quad \forall k \in (0 \dots K-1) \quad (12-3)$$

- قید ۱۳-۳ تضمین می کند که جریان ها قابل تقسیم نیستند.

$$(۱۳-۳)$$

$$f_{uv}^{s_k, s_{k+1}} + f_{vu}^{s_k, s_{k+1}} = bw(s_k, s_{k+1}) \cdot h_{uv}^{s_k, s_{k+1}}, \quad \forall k \in (0 \dots K - 1), (u, v) \in \mathbf{E}'$$

- قید ۱۴-۳ تضمین می کند که هر VNF باید تنها به یک سرور تخصیص داده شود.

$$\sum_{n \in N} h_{s_p n}^{s_k, s_{k+1}} = 1, \quad \forall s_p \in \mathbf{S}, \quad k \in (0 \dots K - 1) \quad (۱۴-۳)$$

- در نهایت، قید ۱۵-۳ اطمینان حاصل می کند که جایابی VNF ها یک گراف متصل تشکیل می دهد.

$$h_{uv}^{s_k, s_{k+1}} = h_{uv}^{s_{k+1}, s_{k+2}}, \quad k \in (0 \dots K - 2), \quad (u, v) \in \mathbf{E}' \quad (۱۵-۳)$$

الگوریتم های اکتشافی حل این مساله در فصل بعد بررسی شده اند.

فصل ۴

روش‌های پیشنهادی و شبیه‌سازی‌ها

۴-۱ مقدمه

در این فصل ابتدا مساله بهینه‌سازی مطرح شده در فصل قبل، با مسائل بهینه‌سازی معروف موجود مقایسه می‌شود و برخی از راه‌حل‌های اکتشافی موجود برای مواجهه با این مساله بیان می‌گردند. سپس دو روش اکتشافی منحصر برای مساله مورد نظر این پژوهش ارائه می‌شوند. عملکرد یکی از روش‌های پیشنهادی بررسی شده و نتایج شبیه‌سازی برای داده‌های تصادفی بررسی می‌شوند.

۴-۲ مساله بسته‌بندی در جعبه‌ها

مساله بسته‌بندی در جعبه‌ها^۱ یا BPP، مساله قراردعی اشیاء دارای حجم‌های مختلف درون تعداد متناهی جعبه با ظرفیت مشخص و یکسان است، به طوری که تعداد جعبه‌های استفاده شده کمینه شود. BPP از مسائل معروف بهینه‌سازی گسسته بوده که دارای کاربردهای زیادی است و به طور طبیعی با مسائل زیادی همچون مساله تخصیص پهنای باند و مساله ایجاد نسخه پشتیبان از فایل‌ها در محیط‌های قابل جابه‌جایی تطبیق می‌یابد.

روش‌های حل BPP را می‌توان به دو دسته کلی آفلاین و آنلاین طبقه‌بندی کرد. روش‌های آفلاین، روش‌هایی هستند که با داشتن اطلاع کامل از لیست اشیاء، آن‌ها را جایابی می‌کنند، در حالی که روش‌های آنلاین به محض

¹bin packing problem

رسیدن اشیا آن‌ها را جایی می‌کنند و از آینده اطلاعی ندارند.

از نظریه پیچیدگی محاسباتی، BPP حتی برای حالت آفلاین نیز یک مساله بهینه‌سازی NP دشوار است. به علاوه، بیان BPP به صورت مساله تصمیم‌گیری (تصمیم این که آیا انتخاب تعداد مشخصی جعبه بهینه است یا خیر) نیز از نوع NP کامل است.

همچنین گفته می‌شود که BPP ویژگی انتقال فاز را از خود نشان می‌دهد. به این معنی که بسته به ویژگی‌های توزیع اندازه‌های اشیا و ظرفیت سطل‌ها، ممکن است پیدا کردن یک راه‌حل بهینه بسیار آسان باشد و یا برعکس، ممکن است یافتن یک راه‌حل بهینه واقعاً به صرف زمان نمایی نیاز داشته باشد. البته این موضوع تنها در صورتی حائز اهمیت است که راه‌حل بهینه دقیق را بخواهیم. از این رو، در حالت کلی روش‌های اکتشافاتی بسیاری برای مواجهه با این مساله ایجاد شده‌اند.

۱-۲-۴ روش‌های حل BPP

تعدادی روش اکتشافی شناخته شده برای حل BPP در ادامه مطرح و مختصراً بررسی خواهند شد.

روش اولین برازش^۲

در روش اولین برازش یا FF، هر شی جدید با وزن مشخص در اولین مکانی که فضای کافی برای جا دادن آن داشته باشد، قرار می‌گیرد. این روش، یک روش آنلاین است. شبه‌کد این روش به صورت ۱-۴ می‌باشد. در بدترین حالت، هر زمان که یک شی جدید وارد می‌شود، نیاز است که یک مکان جدید باز شود. بنابراین حلقه داخلی ۱ - n بار اجرا می‌شود که این امر منجر به پیچیدگی زمانی از مرتبه $O(n^2)$ می‌شود.

روش آخرین برازش نیز همچون روش اولین برازش است، با این تفاوت که هر شی جدید را در آخرین مکان قرار می‌دهد. [۳۵]

روش اولین برازش کاهشی^۳

در روش اولین برازش کاهشی یا FFD، اشیا به ترتیب غیرصعودی از اندازه خود مرتب می‌شوند. شی بعدی در این ترتیب، همواره در اولین مکانی که جا شود، قرار می‌گیرد. از آن‌جا که تمامی وزن‌های اشیا پیش از شروع اجرا روش دانسته فرض می‌شوند، روش Counting Sort یک روش مناسب برای مرتب کردن اشیا خواهد بود. این روش، یک روش آفلاین است. شبه‌کد این روش به صورت ۲-۴ می‌باشد. اگر k بزرگ‌ترین وزن در میان اشیا

^۲First Fit

^۳First Fit Decreasing

Algorithm First-Fit

```
1: for All objects  $i = 1, 2, \dots, n$  do
2:   for All bins  $j = 1, 2, \dots$  do
3:     if Object  $i$  fits in bin  $j$  then
4:       Pack object  $i$  in bin  $j$ .
5:       Break the loop and pack the next object.
6:     end if
7:   end for
8:   if Object  $i$  did not fit in any available bin then
9:     Create new bin and pack object  $i$ .
10:  end if
11: end for
```

شکل ۴-۱: شبه‌کد روش اولین برازش [۳۵]

Algorithm First-Fit-Decreasing

```
1: Sort objects in decreasing order using Counting Sort.
2: Apply First-Fit to the sorted list of objects.
```

شکل ۴-۲: شبه‌کد روش اولین برازش کاهشی [۳۵]

Algorithm Best-Fit

```

1: for All objects  $i = 1, 2, \dots, n$  do
2:   for All bins  $j = 1, 2, \dots$  do
3:     if Object  $i$  fits in bin  $j$  then
4:       Calculate remaining capacity after the object has been added.
5:     end if
6:   end for
7:   Pack object  $i$  in bin  $j$ , where  $j$  is the bin with minimum remaining capacity
   after adding the object (i.e. the object “fits best”).
8:   If no such bin exists, open a new one and add the object.
9: end for

```

شکل ۴-۳: شبه‌کد روش بهترین برازش [۳۵]

باشد، آن‌گاه روش Counting Sort دارای پیچیدگی از مرتبه $O(n + k)$ می‌باشد و روند، به وضوح متاثر از زمان اجرای روش اولین برازش خواهد بود. در نتیجه در این جا نیز پیچیدگی زمانی از مرتبه $O(n^2)$ خواهد بود. [۳۵]

روش بهترین برازش^۴

در روش بهترین برازش یا BF، هر شی جدید در شلوغ ترین مکانی که فضای کافی برای جا دادن آن داشته باشد، قرار می‌گیرد. این روش، یک روش آنالین است. شبه‌کد این روش به صورت ۴-۳ می‌باشد. از آن جایی که در هر گام تمام مکان‌ها مورد بررسی قرار می‌گیرند، زمان اجرای روش از مرتبه $O(n^2)$ خواهد بود. روش بدترین برازش نیز همچون روش بهترین برازش است، با این تفاوت که هر شی جدید را در خلوت‌ترین مکان فعلی قرار می‌دهد. [۳۵]

روش بهترین برازش کاهشی^۵

در روش بهترین برازش کاهشی یا BFD، اشیاء به ترتیب غیرصعودی از اندازه خود مرتب می‌شوند. سپس، روش بهترین برازش روی اشیاء مرتب شده اعمال می‌شود. این روش، یک روش آفلاین است. شبه‌کد این روش به صورت ۴-۴ می‌باشد. [۳۵]

روش برازش بعدی^۶

روش برازش بعدی یا NF، نسخه‌ای تغییر یافته از روش اولین برازش است. شروع این روش، همچون روش اولین برازش، تا پیدا کردن اولین مکان آزاد ادامه پیدا می‌کند، اما دفعه بعد که فراخوانی صورت می‌گیرد، جستجو

^۴Best Fit

^۵Best Fit Decreasing

^۶Next Fit

Algorithm Best-Fit-Decreasing

- 1: Sort objects in decreasing order using Counting Sort.
 - 2: Apply Best-Fit to the sorted list of objects.
-

شکل ۴-۴: شبه‌کد روش بهترین برازش کاهشی [۳۵]

Algorithm Next-Fit

- 1: **for** All objects $i = 1, 2, \dots, n$ **do**
 - 2: **if** Object i fits in current bin **then**
 - 3: Pack object i in current bin.
 - 4: **else**
 - 5: Create new bin, make it the current bin, and pack object i .
 - 6: **end if**
 - 7: **end for**
-

شکل ۵-۴: شبه‌کد روش برازش بعدی [۳۵]

از مکانی که جایابی آن‌جا متوقف شده بود شروع می‌شود، نه از ابتدا. این روش، یک روش آنالین است. شبه‌کد این روش به صورت ۵-۴ می‌باشد. از آن‌جا که جایابی یک شی می‌تواند در زمان ثابت انجام شود، زمان اجرای روش متأثر از زمان اجرای حلقه است و زمان اجرایی از مرتبه $O(n)$ دارد. [۳۵]

روش برازش بعدی کاهشی^۷

در روش برازش بعدی کاهشی یا NF، اشیاء به ترتیب غیرصعودی از اندازه خود مرتب می‌شوند. سپس، روش برازش بعدی روی اشیاء مرتب شده اعمال می‌شود. این روش، یک روش آفلاین است. شبه‌کد این روش به صورت ۶-۴ می‌باشد.

از آن‌جا که زمان اجرای روش برازش بعدی دارای مرتبه $O(n)$ می‌باشد، عامل تعیین‌کننده روش Counting Sort خواهد بود که زمان اجرای آن از مرتبه $O(n + k)$ است که k بزرگترین وزن در میان اشیاء است. [۳۵]

روش باقی‌مانده بیشینه^۸

در روش باقی‌مانده بیشینه یا MR، ابتدا مکان با بیش‌ترین ظرفیت باقی‌مانده تعیین می‌شود. اگر شی فعلی در آن مکان جای گیرد، همان مکان به عنوان مکان شی انتخاب می‌شود و در غیر این صورت، شی در یک مکان

⁷Next Fit Decreasing

Max Rest^۸

Algorithm Next-Fit-Decreasing

- 1: Sort objects in decreasing order using Counting Sort.
- 2: Apply Next-Fit to the sorted list of objects.

شکل ۴-۶: شبه‌کد روش برازش بعدی کاهشی [۳۵]

Algorithm Max-Rest

- 1: **for** All objects $i = 1, 2, \dots, n$ **do**
- 2: Determine $k = \min\{i \mid c_i = \min_{j=1}^{j=m} c_j\}$, the index of the bin with maximum remaining capacity.
- 3: **if** Object i fits in bin k **then**
- 4: Pack object i in bin k .
- 5: **else**
- 6: Create new bin and pack object i .
- 7: **end if**
- 8: **end for**

شکل ۴-۷: شبه‌کد روش باقی مانده بیشینه [۳۵]

جدید جایابی می‌شود. شبه‌کد این روش به صورت ۴-۷ می‌باشد. با استفاده از یک روش ساده، تعیین مکان با بیش‌ترین ظرفیت باقی مانده منجر به پیچیدگی با فاکتور $O(n)$ می‌شود. در نتیجه، بدترین حالت زمان اجرای روش دارای فاکتور $O(n^2)$ خواهد بود.

با یک بررسی جزئی تر مشخص می‌شود که در برخی از روش‌های مذکور، جهت تعیین مکان جایابی، می‌توان از یک صف اولویت استفاده کرد. در این حالت، می‌توان هر یک از مکان‌های جایابی را طی مدت زمانی ثابت، معین نمود. یعنی جایابی هر شی (چه در مکان جدید و چه در مکان فعلی) به افزودن یا به روزرسانی یک جز از صف نیاز دارد که می‌تواند در زمان $O(\log n)$ انجام شود. بنابراین، نسخه ارتقا یافته روش‌ها، زمان اجرایی از مرتبه $O(n \log n)$ خواهند داشت. مثلاً برای روش باقی مانده بیشینه به صورت ۴-۸ می‌باشد. [۳۵]

Algorithm Max-Rest-Priority-Queue

```

1: for All objects  $i = 1, 2, \dots, n$  do
2:   if Object  $i$  fits in top-most bin of the priority queue then
3:     Remove top-most bin from queue.
4:     Add object  $i$  to this bin.
5:     Push updated bin to queue.
6:   else
7:     Create new bin and pack object  $i$ .
8:   end if
9: end for

```

شکل ۴-۸: شبه‌کد روش باقی مانده بیشینه با صف اولویت بندی [۳۵]

۴-۲-۲ BPP با اندازه‌های متفاوت

در ادامه به بیان مساله BPP تک بعدی دارای اندازه‌های متفاوت^۹ یا VSBPP می‌پردازیم که به طور قراردادی به صورت زیر تعریف می‌شود:

یک مجموعه از تعدادی شی با وزن‌های مشخص و همچنین تعدادی جعبه از چندین نوع مختلف فرض می‌شوند. هر نوع از جعبه‌ها، دربردارنده تعدادی جعبه یکسان با ظرفیت مشخص و هزینه ثابت می‌باشد. هدف VSBPP این است که هر شی از مجموعه اشیاء را درون یک جعبه به نحوی جایابی کند که مجموع وزن‌ها در هر جعبه از ظرفیت آن جعبه بیشتر نشده و هزینه کل کمینه شود. واضح است در حالت خاص که تمام جعبه‌ها یکسان باشند، مساله تبدیل به BPP تک‌بعدی خواهد شد. همان‌طور که گفته شد، BPP، NP دشوار است و نتیجتاً VSBPP نیز غیر قابل پیگیری خواهد بود.

این مساله، در متون به عنوان مساله برش سهام^{۱۰} با اندازه‌های مختلف سهام نیز شناخته می‌شود. جعبه‌ها همان طول‌های استاندارد موادی مانند کاغذ، چوب یا کابل‌های برق هستند و اشیاء، همان طول‌هایی هستند که باید از آن طول‌های استاندارد بریده شوند. هنگامی که بیش از یک طول استاندارد در دسترس باشد، نسخه دارای سایز متغیر مسئله به دست می‌آید.

در مساله بارگیری کامیون^{۱۱}، VSBPP وقتی بوجود می‌آید که وزن تنها بعدی باشد که در نظر گرفته می‌شود

^۹Variable Sized Bin Packing Problem

^{۱۰}cutting stock problem

^{۱۱}truck loading problem

و همچنین بیش از یک کامیون از هر نوع یا سایز در دسترس باشد.

همچنین، VSBPP می‌تواند به عنوان یک حالت خاص از دو مساله بهینه‌سازی ترکیبی در نظر گرفته شود: مساله اندازه ناوگان و مسیریابی ترکیبی وسایل نقلیه^{۱۲} (FSMVRP) و مساله موقعیت یابی تک منبعی^{۱۳} (SCFLP). FSMVRP، مساله تصمیم‌گیری در مورد ترکیب و اندازه ناوگانی از وسایل نقلیه و ساخت یک مجموعه از مسیرهای مرتبط برای این وسایل نقلیه است، به طوری که به مجموعه‌ای از پیش تعیین شده از مشتریان با تقاضاهای دانسته خدمات ارائه دهند. هدف از این مساله، به حداقل رساندن مجموع هزینه‌های ثابت (ناشی از استفاده از وسیله نقلیه) و هزینه‌های مسیریابی (مربوط به حرکات بین انبار و محل‌های مشتریان) است. بدیهی است اگر هزینه‌های متغیر سفر نسبت به هزینه‌های ثابت قابل چشم‌پوشی باشند، FSMVRP معادل VSBPP می‌شود.

SCFLP، شامل تخصیص مجموعه‌ای از مشتریان با تقاضاهای دانسته به مجموعه‌ای از تسهیلات است، به طوری که هر مشتری بدون نقض ظرفیت‌های تسهیلات مورد استفاده، به یکی از تسهیلات اختصاص داده شود. هدف این مساله به حداقل رساندن مجموع هزینه‌های راه‌اندازی تسهیلات انتخابی و هزینه‌های اتصال مشتریان به آن تسهیلات است. در این‌جا نیز مجدداً، اگر هزینه‌های اتصال قابل چشم‌پوشی باشد، SCFLP معادل VSBPP می‌شود [۱۹].

پیشینه تحقیقاتی مرتبط با VSBPP در مقایسه با BPP نسبتاً کم است. تاکنون، چندین پژوهش، راه‌حل‌های تقریبی برای VSBPP و حالت‌های تغییر یافته آن بررسی کرده‌اند و حدود بدترین حالت مطلق و/یا تقریبی پیشنهاد داده و تحلیل نموده‌اند. تعاریف این حدود در بخش پیوست‌ها آمده‌اند.

در پژوهش [۲۲]، برای حل سه حالت خاص از VSBPP و شرایطی که هزینه هر واحد از حجم جعبه‌ها با افزایش حجم جعبه افزایش پیدا نکند، دو روش حریصانه^{۱۴} پیشنهاد می‌شود که برگرفته از روش‌های شناخته شده FFD و BFD می‌باشند. حالت‌های در نظر گرفته شده از نظر تقسیم‌پذیر بودن یا نبودن وزن اشیاء و یا ظرفیت جعبه‌ها تفکیک شده‌اند. تقسیم‌پذیر بودن در این‌جا به این معنی است که مثلاً در مورد جعبه‌ها، هر ظرفیت بالاتر در مجموعه جعبه‌ها، به طور دقیق قابل تقسیم توسط اندازه‌های ظرفیت‌های پایین‌تر در مجموعه جعبه‌ها باشد. گفته شده است که تقسیم‌پذیر بودن یا نبودن اندازه‌های اشیاء و جعبه‌ها یکی از مواردی است که روی دقت پاسخ‌های استخراج شده از اعمال روش‌های مختلف برای حل VSBPP موثر است. برای حالتی که هم اندازه‌های

¹² fleet size and mix vehicle routing problem

¹³ single source capacitated location problem

¹⁴ greedy

اشیا و هم جعبه‌ها تقسیم پذیر باشند، روش‌های ارائه شده به پاسخ بهینه دست پیدا می‌کنند. در پژوهش [۱۳]، سه روش تقریبی NFL، FFDLR و FFDLS ارائه می‌شوند که حدود عملکرد بدترین حالت تقریبی آن‌ها به ترتیب ۲، ۲/۳ و ۳/۴ می‌باشند. پژوهش [۳۰] یک روش مبتنی بر شاخه و حد برای حل نمونه‌هایی از VSBPP با حداکثر ۵۰۰ شی و ۳ یا ۵ نوع جعبه پیشنهاد می‌کند. در این مقاله، هزینه هر جعبه برابر با ظرفیت آن و تعداد جعبه‌های هر نوع، برابر با تعداد کل اشیا در نظر گرفته می‌شود. در هر یک از پژوهش‌های [۶] و [۵]، روش‌های تولید ستون^{۱۵} نسبتاً کارآمدی برای حل نمونه‌های BPP و VSBPP با حداکثر ۵۰۰ شی مورد استفاده قرار می‌گیرند. در این پژوهش‌ها، هدف به حداقل رساندن مجموع ظرفیت‌های جعبه‌های مورد استفاده در پاسخ است. روش‌های تولید ستون همچنین در مقاله [۳۱] نیز برای حل یک نسخه دو بعدی از VSBPP توسعه داده شده و مورد استفاده قرار گرفته است.

۳-۴ مساله یافتن کوتاه‌ترین مسیر

مساله کوتاه‌ترین مسیر^{۱۶} یکی از اساسی‌ترین مسائل بهینه‌سازی شبکه است. این مساله در عمل به وجود آمد و در بسیاری از روش‌های بهینه‌سازی شبکه به عنوان زیرمساله مطرح می‌شود. مساله کوتاه‌ترین مسیر در واقع مساله یافتن مسیری بین دو گره است، به گونه‌ای که مجموع وزن یال‌های تشکیل‌دهنده آن کمینه شود. این مساله برای یافتن مسیرهای میان مکان‌های فیزیکی از قبیل راه‌های عبور و مرور در نقشه‌های اینترنتی مانند نقشه گوگل استفاده می‌شود.

مهم‌ترین روش‌ها برای حل این مسئله عبارتند از:

- روش بلمن-فورد:^{۱۷} مساله یافتن کوتاه‌ترین مسیر از مبدا واحد را در حالتی حل می‌کند که وزن یال‌ها منفی نیز می‌تواند باشد.
- روش فلوید-وارشال:^{۱۸} مساله یافتن کوتاه‌ترین مسیر بین هر دو راس را حل می‌کند.

¹⁵column generation

¹⁶shortest path problem

¹⁷Bellman-Ford algorithm

¹⁸Floyd-Warshall algorithm

• روش ویتربی: ^{۱۹} مساله کوتاهترین مسیر تصادفی را با در نظر گرفتن یک وزن احتمالی روی هر گره حل می‌کند.

• روش دایجسترا: ^{۲۰} مساله یافتن کوتاهترین مسیر بین دو راس، از مبدا واحد و به مقصد واحد را حل می‌کند که در آن وزن تمامی یال‌ها غیر منفی در نظر گرفته می‌شود.

روش دایجسترا یکی از مهم‌ترین روش‌های ارائه شده برای حل مساله کوتاهترین مسیر می‌باشد. یک پیاده‌سازی از این روش با دنباله فیبوناچی که در [۱۲] آمده است، دارای پیچیدگی زمانی از مرتبه $O(m + n \log n)$ می‌باشد که در آن n تعداد گره‌ها و m تعداد یال‌های گراف شبکه است [۲]. مراحل اجرای روش دایجسترا مطابق زیر می‌باشد:

۱. راس مبدا انتخاب می‌شود.
 ۲. مجموعه S ، شامل رئوس گراف، تعریف می‌شود. در شروع، این مجموعه تهی بوده و با پیشرفت الگوریتم، رئوسی که کوتاهترین مسیر به آن‌ها یافت شده است را در بر می‌گیرد.
 ۳. راس مبدا با اندیس صفر در داخل S قرار می‌گیرد.
 ۴. برای رئوس خارج از S ، اندیسی معادل «طول یال به علاوه اندیس راس قبلی» در نظر گرفته می‌شود. اگر راس خارج از مجموعه دارای اندیس باشد، اندیس جدید کمترین مقدار میان اندیس قبلی و «طول یال به علاوه اندیس راس قبل» می‌باشد.
 ۵. از رئوس خارج مجموعه، راسی با کمترین اندیس انتخاب شده و به مجموعه S اضافه می‌گردد.
 ۶. این کار دوباره از مرحله ۴ ادامه داده می‌شود تا راس مقصد وارد مجموعه S شود.
- در پایان اگر راس مقصد دارای اندیس باشد، اندیس آن نشان دهنده مسافت بین مبدا و مقصد می‌باشد. در غیر این صورت هیچ مسیری بین مبدا و مقصد موجود نمی‌باشد [۱]. شبه‌کد یک روش دایجسترا که فاصله از یک مبدا را تا تمام گره‌های یک گراف محاسبه می‌کند، مطابق شکل ۴-۹ می‌باشد.

¹⁹Viterbi algorithm

²⁰Dijkstra's algorithm

Algorithm 4-1 Dijkstra

Input G : Graph of the network

Input S : Source node

Q : the set of all nodes in G

for each vertex v in Graph: **do**

$\text{dist}[v] := \text{infinity}$

$\text{previous}[v] := \text{undefined}$

end for

$\text{dist}[S] := 0$

while Q is not empty: **do**

$u := \text{node in } Q \text{ with smallest } \text{dist}[]$

 remove u from Q

for each neighbor v of u **do**

$\text{alt} := \text{dist}[u] + \text{dist-between}(u,v)$

if $\text{alt} < \text{dist}[v]$ **then**

$\text{dist}[v] := \text{alt}$

$\text{previous}[v] := u$

end if

end for

end while

return $\text{previous}[]$

شکل ۴-۹: شبه کد الگوریتم دایجسترا [۱۵]

در سیستم مدل ارائه شده در فصل سوم، لازم است که پس از جایابی VNF های زنجیره خدمات، اتصال میان آن‌ها نیز از مسیری که کمترین هزینه را داشته باشد، برقرار گردد. لازم به ذکر است که فرض بر این است که تقسیم جریان در شبکه اتفاق نمی افتد و تمام جریان بین دو نقطه از یک مسیر عبور می کند.

۴-۴ روش‌های اکتشافی پیشنهادی

با بررسی دقیق سیستم مدل مطرح شده در این پژوهش، می توان گفت مساله اصلی از دو زیر مساله تشکیل شده است. اولین زیر مساله، مساله جایابی VNF ها روی سرورهای شبکه است و زیر مساله دوم، مسیریابی از میان این VNF ها است.

با مقایسه زیر مساله اول با VSBPP، مشخص می‌شود که هر سرور در نقش یک جعبه بوده و VNF‌ها همان اشیای مذکور در VSBPP هستند. نتیجتاً جایابی اشیای درون جعبه، معادل مساله جایابی VNF بر روی سرورهای شبکه خواهد بود.

سپس با مقایسه زیر مساله دوم با مساله یافتن کوتاه‌ترین مسیر، تشابه این دو مساله نیز آشکار می‌گردد. به این صورت که باید بین هر دو VNF متوالی در زنجیره خدمات که جایابی شده‌اند، مسیری با کمترین هزینه برقرار شود. هدف نهایی این است که هزینه کل استفاده از سرورها و لینک‌های موجود در ساختار شبکه کمینه شود. در ادامه، دو روش اکتشافی برای حل مساله جایابی و مسیریابی VNF‌ها ارائه شده‌است. جهت یادآوری بیان می‌شود که S زنجیره خدمات یا همان مجموعه VNF‌های دارای ترتیب است که باید جایابی شوند. k امین VNF با s_k نشان داده می‌شود و به اندازه $D(s_k)$ ظرفیت سرور را اشغال می‌کند. مجموعه سرورها با N نشان داده می‌شود. با در نظر گرفتن منافع فراهم‌آورنده سرویس و فراهم‌آورنده ابری، هر سرور $n \in N$ دارای هزینه $A_n R_n$ می‌باشد و ظرفیت CPU آن با C_n نشان داده می‌شود. همچنین، بدون از دست دادن کلیت مساله فرض می‌شود که ظرفیت و هزینه سرورها و اندازه VNF‌ها اعداد صحیح مثبت هستند و اندازه بزرگترین VNF کمتر از ظرفیت بزرگترین سرور است.

۴-۱-۴ روش SSP محور

این روش مبتنی بر مساله مجموع زیرمجموعه یا SSP^{۲۱} می‌باشد. در این روش، ابتدا مجموعه‌های سرورها و VNF‌ها و گراف توپولوژی شبکه به عنوان ورودی دریافت می‌شوند. سپس سرورها به ترتیب غیرصعودی از ظرفیت خود مرتب می‌شوند و یک مجموعه از VNF‌هایی که تا کنون به هیچ سرورری تخصیص داده نشده‌اند، با نام \bar{S} تعریف می‌گردد. بدیهی است که در ابتدا، \bar{S} برابر با مجموعه تمام VNF‌ها می‌باشد. [۱۸]

اولین سرور از مجموعه سرورهای مرتب شده انتخاب می‌گردد و برای تکمیل آن، یک SSP بر روی اعضای \bar{S} حل می‌شود و سرور مذکور با مجموعه پاسخ، یعنی مجموعه VNF‌های به دست آمده از حل SSP، تکمیل می‌گردد. توضیحات تکمیلی در ارتباط با SSP و نحوه حل آن، در پیوست آورده شده‌است.

پس از تکمیل اولین سرور، به روز رسانی‌ها اجرا می‌شوند، به این صورت که سرور تکمیل شده از لیست سرورها و مجموعه پاسخ از \bar{S} حذف می‌شوند. سپس سرور بعدی از لیست برای تکمیل در نظر گرفته می‌شود.

²¹Subset Sum Problem

این کار تا زمانی ادامه پیدا می‌کند که هیچ VNF ای در \bar{S} باقی نمانده باشد.

پس از جابجایی همه VNF ها، در صورتی که سرورهای میزبان هر دو VNF متوالی متفاوت باشند، از روش دایجسترا برای یافتن کوتاه‌ترین مسیر میان آن دو VNF استفاده می‌شود. همان‌طور که در فصل سوم نیز بیان شد، فرض بر این است که مسیریابی‌های مربوط به درون یک سرور، در مساله حاضر در نظر گرفته نمی‌شوند. شبه‌کد این روش به صورت ۱۰-۴ است.

Algorithm 4-2 SSP Based

Input S : Set of VNFs to be accommodated into the servers

Input N : Set of available servers

Input G : Topology graph of network

Sort the servers in N according to non-increasing order of their capacities

\bar{S} : Set of unpacked VNFs

Set $\bar{S} = S$

while $\bar{S} \neq \emptyset$ **do**

 Solve a SSP to load a subset of \bar{S} into the first server of the set N , naming n

 Define S_n as the set of selected VNFs by solving SSP

$N := N \setminus n$

$\bar{S} := \bar{S} \setminus S_n$

end while

if $\bar{S} = \emptyset$ **then**

for all $s_k \in S$ **do**

if selected servers of s_k and s_{k+1} are different **then**

$dijkstra(G, s_k, s_{k+1})$

end if

end for

end if

شکل ۱۰-۴: شبه‌کد الگوریتم دایجسترا [۱۵]

۲-۴-۴ روش BFD تطبیقی

روش بهترین برازش کاهشی تطبیقی^{۲۲} یا ABFD، از روش شناخته شده BFD استفاده می‌کند که پیش‌تر راجع به آن توضیح داده شد و برای BPP کلاسیک عملکرد قابل قبولی دارد.

در این روش، ابتدا مجموعه‌های سرورها و VNF‌ها و گراف توپولوژی شبکه به عنوان ورودی دریافت می‌شوند. سپس VNF‌ها به ترتیب غیر صعودی از اندازه خود و سرورها به ترتیب غیر نزولی از میزان هزینه هر واحد از ظرفیت خود مرتب می‌گردند. همچنین یک مجموعه از VNF‌هایی که تا کنون به هیچ سرورری تخصیص داده نشده‌اند، با نام \bar{S} و یک مجموعه از سرورهایی که تا کنون انتخاب شده‌اند با نام K تعریف می‌شوند. بدیهی است که در ابتدا، \bar{S} برابر با مجموعه تمام VNF‌ها و K برابر با تهی می‌باشند.

برای جایابی اولین VNF از مجموعه \bar{S} ، ابتدا سعی بر آن است که به بهترین سرورری که تا کنون گشوده شده، یعنی بهترین عضو K ، تخصیص داده شود. برای تعیین بهترین سرور، تابع معیاری بر اساس ملاحظات تابع هدف ارائه شده در رابطه ۳-۴ تعریف می‌شود. بهترین سرور در این روش، سرورری است که تابع معیاری که ظرفیت خالی سرورها را محاسبه می‌کند، حداکثر نماید. اگر VNF نتواند روی یکی از سرورهای از پیش انتخاب شده جای گیرد، آن گاه اولین سرور از مجموعه مرتب‌شده سرورهای انتخاب‌نشده که ظرفیت کافی برای قراردعی VNF مذکور را داشته باشد، انتخاب می‌شود و VNF روی آن قرار می‌گیرد.

پس از جایابی اولین VNF، به روزرسانی‌ها اجرا می‌شوند، به این صورت که سرور انتخاب شده به K اضافه می‌شود و VNF جایابی شده از \bar{S} حذف می‌شود. سپس VNF بعدی از مجموعه \bar{S} برای جایابی در نظر گرفته می‌شود. این کار تا زمانی ادامه پیدا می‌کند که هیچ VNF‌ای در \bar{S} باقی نمانده باشد.

یک موضوع که مختص به VSBPP است و در BPP کلاسیک که در آن جعبه‌ها مشابه هستند ظاهر نمی‌شود، این که در مواقع لزوم جعبه جدید طبق چه معیاری انتخاب شود. در این‌جا، سرورهای جدید مطابق ترتیبی غیر نزولی از هزینه هر واحد از ظرفیت انتخاب می‌شوند. انتخاب سرورها با این معیار، معمولاً به نتایج خوبی منجر می‌شود، ولی هنگام جایابی آخرین VNF‌ها ممکن است عملکرد کمی تنزل پیدا کند، حتی اگر هزینه هر واحد از ظرفیت سرور انتخاب شده پایین باشد. اتفاقی که ممکن است رخ بدهد این است که هنگامی یک سرور جدید برای یکی از VNF‌های پایان لیست انتخاب و گشوده می‌شود، سرور انتخاب‌شده ممکن است ظرفیتی بسیار بیشتر از آن VNF داشته باشد و VNF‌های اندکی نیز برای بهره بردن از این ظرفیت باقی مانده باشند.

²² adaptive best fit decreasing

برای حل این موضوع، یک انتخاب مناسب می‌تواند سروری با نسبت هزینه به ظرفیت بالاتر، اما هزینه کل پایین‌تر باشد. در نتیجه می‌توان بعد از پردازش‌های اولیه، یک مرحله کنترلی به روش اضافه کرد که با ارزیابی چنین تبادلهای احتمالی بین سرورها، پاسخ روش را بهبود بخشد. این مرحله متناوباً هر یک از سرورهای انتخاب شده را بررسی می‌کند. سپس در صورت وجود یک سرور انتخاب نشده با هزینه کل کمتر و ظرفیت بیشتر یا مساوی با ظرفیت استفاده شده از سرور انتخاب شده، VNFهایی که روی سرور قبلی جایابی شده‌اند را به سرور جدید منتقل می‌کند. [۹]

در این روش نیز پس از جایابی همه VNFها، در صورتی که سرورهای میزبان هر دو VNF متوالی متفاوت باشند، از روش دایجسترا برای یافتن کوتاه‌ترین مسیر میان آن دو VNF استفاده می‌شود. شبه‌کد این روش به صورت ۴-۱۱ است.

۴-۵ شبیه‌سازی و نتایج

در ادامه، ابتدا مفروضاتی که برای طراحی سیستم مدل در نظر گرفته شده‌اند و همچنین محیط سخت افزاری و نرم افزاری شبیه‌سازی‌ها معرفی می‌شوند، سپس نتایج مقایسه‌ای حاصل از اجرای روش بهینه و الگوریتم اکتشافی ABFD ارائه می‌گردند.

۴-۵-۱ مفروضات شبیه‌سازی

برای اجرا شبیه‌سازی‌ها، از رایانه‌ای با ویندوز ۱۰، پردازنده اینتل ۷ هسته‌ای و حافظه ۱۶ گیگابایتی استفاده شده‌است. روش‌های مورد بررسی، در نرم‌افزار متلب ۲۰۱۹ و با بهره‌گیری از Toolbox Mosek شبیه‌سازی شده‌اند.

همان‌طور که در فصل سوم نیز به آن اشاره گردید، سوئیچ‌ها، سرورها و VNFها به عنوان گره‌های شبکه در نظر گرفته می‌شوند و فرض می‌شود که بین تمام سوئیچ‌ها و سرورها و همچنین تمام سرورها و VNFهای زنجیره خدمات، لینک وجود دارد. در نتیجه نمایش گراف شبکه به صورت شکل ۴-۱۲ خواهد بود. در این شکل، دو عدد سوئیچ، دو عدد سرور و یک زنجیره خدمات شامل سه عدد VNF موجود می‌باشند. لازم به ذکر است که در

بعضی از شبیه‌سازی‌ها، از جمله برای تحلیل مقیاس‌پذیری سیستم، تعداد سرورهای شبکه با حفظ کلیت ساختار تغییر خواهد کرد.

فرض می‌شود که ترتیب VNF‌ها درون زنجیره خدمات مشخص و ثابت است. همچنین فرض می‌شود که پذیرش زنجیره خدمات به درون شبکه نیز از قبل صورت گرفته‌است.

برای هر یک از سرورها مقدار مشخصی ظرفیت CPU در نظر گرفته شده‌است و به هر یک از لینک‌های شبکه، مقدار مشخصی پهنای باند اختصاص داده شده که به صورت وزن لینک‌ها در گراف توپولوژی شبکه نشان داده می‌شوند. پهنای باند لینک متصل‌کننده سوئیچ‌ها، به میزان چشم‌گیری بیشتر از پهنای باند لینک‌های واصل سرورها و سوئیچ‌ها است.

۲-۵-۴ نتایج شبیه‌سازی

در مجموع، جهت استخراج هر یک از نمودارهایی که در ادامه خواهند آمد، شبیه‌سازی‌ها صد مرتبه تکرار شده و سپس از نتایج آن‌ها میانگین گرفته شده‌است. جهت مقایسه عملکرد الگوریتم اکتشافی پیشنهادی با حالت بهینه، از تعدادی معیار ارزشیابی استفاده می‌شود که در ادامه معرفی می‌شوند.

در ابتدا، الگوریتم‌ها از نظر تابع هدف ارائه شده در معادله ۳-۴ مقایسه می‌شوند. به این منظور، تاثیر افزایش تعداد زنجیره‌های خدمات پذیرفته شده درون شبکه و همچنین تاثیر افزایش ازدحام قابل تحمل توسط هر یک از لینک‌های شبکه بر روی هزینه کل بررسی می‌گردد. سپس، تعداد سرورهای مورد استفاده برای تخصیص یک زنجیره خدمات مورد بررسی قرار می‌گیرد. در نهایت، مقیاس‌پذیری الگوریتم‌ها از نظر زمان اجرای الگوریتم ارزیابی می‌گردد.

شکل ۴-۱۳، میانگین هزینه کل جایابی و مسیریابی توابع شبکه را بر حسب تعداد زنجیره‌های خدمات پذیرفته شده درون شبکه نشان می‌دهد.

مشاهده می‌شود که برای هر دو الگوریتم، با افزایش تعداد زنجیره‌های خدمات، هزینه کل نیز افزایش می‌یابد. این افزایش هزینه، با توجه به افزایش نیاز به ظرفیت CPU سرورها و پهنای باند لینک‌ها توسط زنجیره‌های جدید قابل توجیه است.

همچنین، همان‌طور که انتظار می‌رفت الگوریتم MIP با کمترین هزینه، قادر به ارائه راه‌حل بهینه می‌باشد.

شکل ۴-۱۴، تاثیر افزایش اندازه بافر لینک‌های شبکه را بر روی میانگین هزینه کل جایابی و مسیریابی توابع

شبکه نشان می‌دهد.

در این جا فرض بر آن است که درون شبکه، یک زنجیره خدمات وجود دارد. مشاهده می‌شود که هر دو الگوریتم اکتشافی و بهینه، رفتار کاهش هزینه نسبت به افزایش اندازه بافر لینک‌ها را از خود نشان می‌دهند و در عین حال، روش بهینه در هر نقطه از نمودار دارای هزینه کل کمتری نسبت به الگوریتم اکتشافی می‌باشد. اندازه بافر هر لینک، تعیین‌کننده میزان ازدحام قابل تحمل توسط آن لینک می‌باشد. علت کاهش هزینه این است که با افزایش اندازه بافر، لینک قادر به تحمل ازدحام بالاتری خواهد بود. این افزایش اندازه بافر، با بالا بردن قدرت تحمل ازدحام هر لینک، بر میزان استفاده از هر یک از لینک‌های شبکه تاثیر می‌گذارد و در نهایت منجر به کاهش هزینه کل جابجایی و مسیریابی VNF ها می‌گردد.

در ادامه، نمودار ۴-۱۵ به بررسی تاثیر افزایش تقاضای منبع بر روی میانگین تعداد سرورهای اختصاص یافته به یک زنجیره خدمات می‌پردازد.

در این جا نیز فرض بر آن است که درون شبکه، یک زنجیره خدمات وجود دارد. با افزایش تقاضای CPU از جانب زنجیره خدمات در هر دو الگوریتم بهینه و اکتشافی، تعداد سرورهای اختصاص یافته به آن زنجیره خدمات افزایش پیدا می‌کند. مجدداً مشاهده می‌شود که الگوریتم بهینه از سرورهای کمتری جهت ارائه خدمت مشابه استفاده می‌کند.

در شکل ۴-۱۶، مقیاس پذیری الگوریتم‌ها با بررسی زمان اجرای الگوریتم بر حسب افزایش تعداد گره‌ها در زیرساخت ابری، ارزیابی شده‌است.

دلیل افزایش زمان اجرای الگوریتم‌ها بر حسب افزایش تعداد گره‌ها در زیرساخت ابری این است که با افزایش اندازه زیرساخت ابری، هر الگوریتم باید تعداد حالات بیشتری را در فضای جستجو بررسی می‌کند که این امر منجر به اجرای تعداد بیشتری عملیات می‌گردد.

باید توجه داشت که با افزایش اندازه زیرساخت ابری، زمان لازم برای اجرای الگوریتم MIP دارای رشد نمایی خواهد بود و ممکن است حتی در محورهای مشابه با سایر الگوریتم‌های اکتشافی مقیاس بندی نشود. نتیجتاً الگوریتم MIP نمی‌تواند در یک آزمایش آنلاین و در مقیاس بزرگ مورد استفاده قرار گیرد، بلکه فقط برای آزمایش‌های در مقیاس کوچک و عمدتاً به منظور مقایسه استفاده می‌گردد.

این افزایش زمان، برای الگوریتم ABFD نسبت به روش MIP بسیار کمتر است، که این عامل موجب مقیاس‌پذیری این الگوریتم و کارا بودن آن به هنگام توسعه شبکه می‌گردد.

Algorithm 4-3 Adapted BFD

Input S : Set of VNFs to be accommodated into the servers

Input N : Set of Servers available to load the VNFs

Input G : Topology graph of network

Sort the VNFs in S according to non-increasing order of their volumes

Sort the servers in N according to non-decreasing order of ratio $A_n R_n / C_n$

\bar{S} : Set of unpacked VNFs

K : Set of selected servers

$\bar{S} = \text{sorted } S$

$K = \{\emptyset\}$

while $\bar{S} \neq \emptyset$ **do**

if the first VNF in \bar{S} , naming s , can be accommodated into a server in K **then**

 Accommodate s into the best server of the set K , naming n

else

 Accommodate s into n' , where n' is the first server in the ordered list $N \setminus K$

$K := K \cup n'$

$\bar{S} := \bar{S} \setminus s$

end if

end while

for all $k \in K$ **do**

for all $m \in N \setminus K$ **do**

$U_k = \sum_{s \text{ loaded in } k} D_s$

if $C_m > U_k$ and $A_m R_m < A_k R_k$ **then**

 Move all the items from k to m

$K = K \setminus k \cup m$

end if

end for

end for

if $\bar{S} = \emptyset$ **then**

for all $s_k \in S$ **do**

if selected servers of s_k and s_{k+1} are different **then**

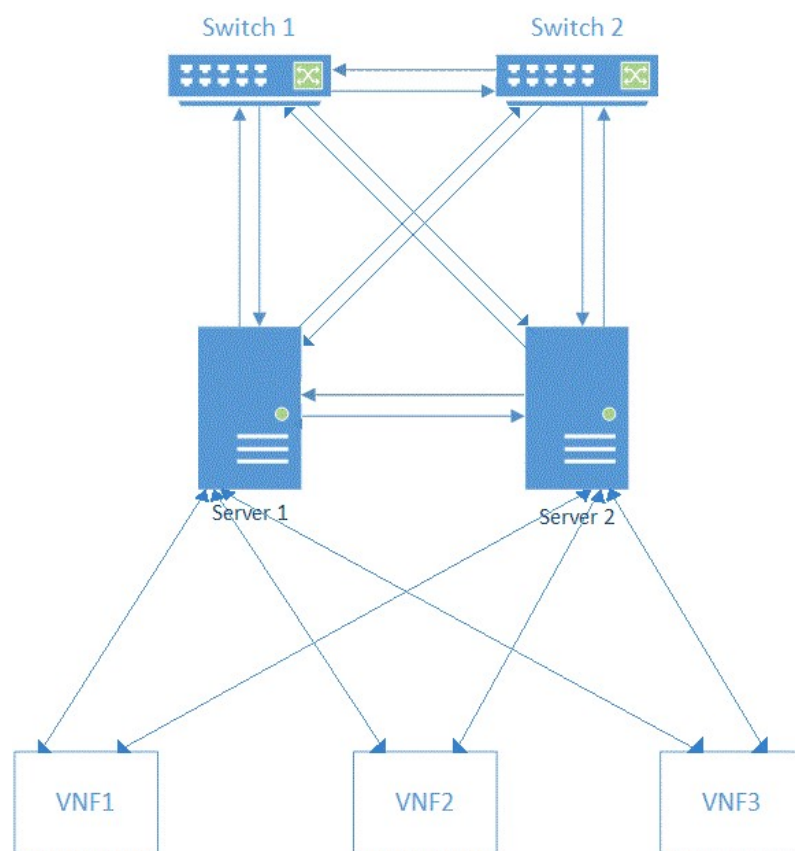
$dijkstra(G, s_k, s_{k+1})$

end if

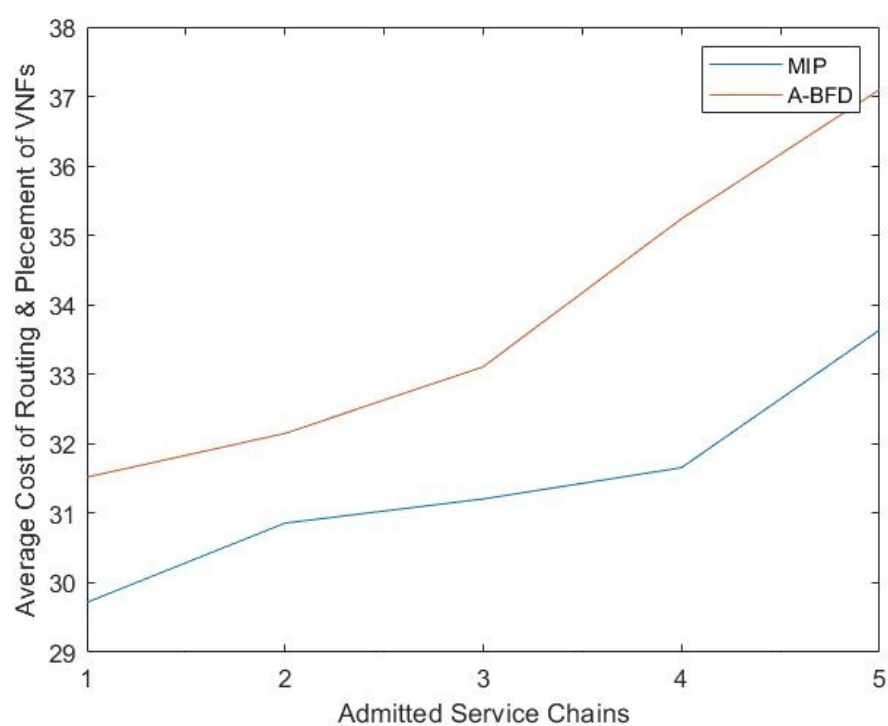
end for

end if

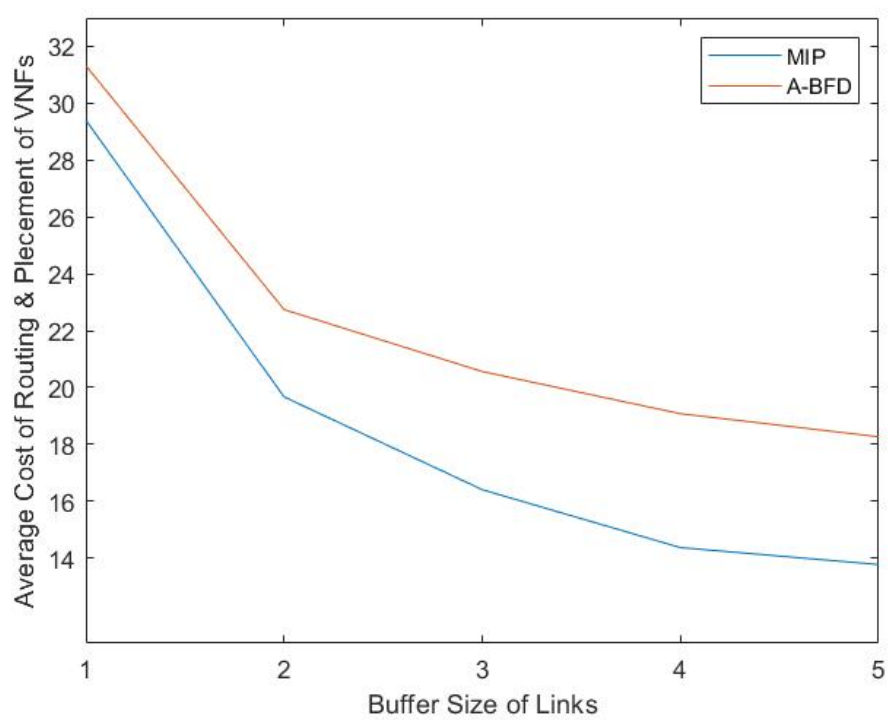
شکل ۴-۱۱: شبه کد الگوریتم دایجسترا [۱۵]



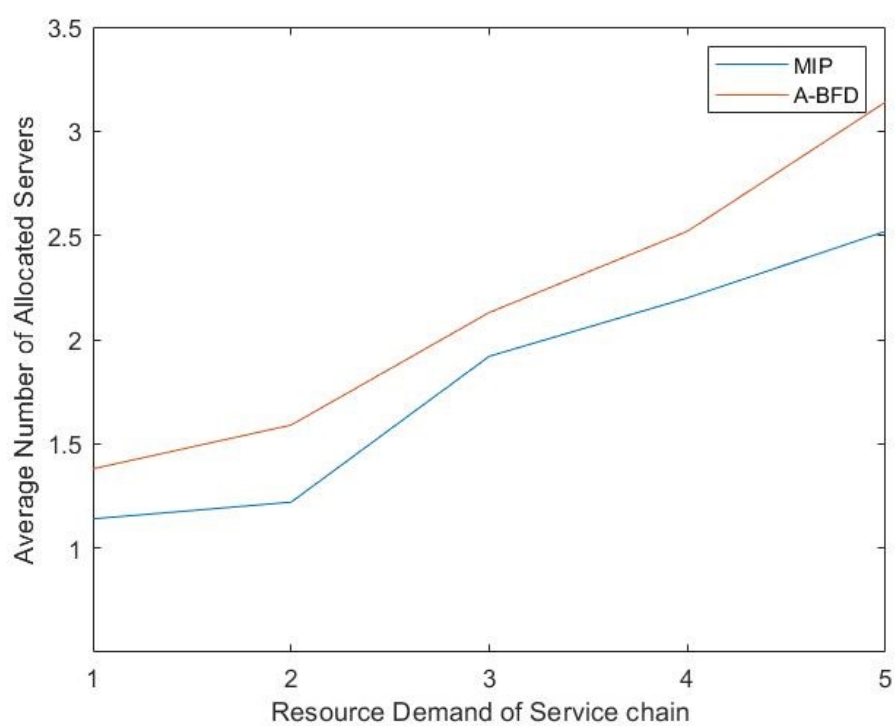
شکل ۴-۱۲: گراف توپولوژی شبکه



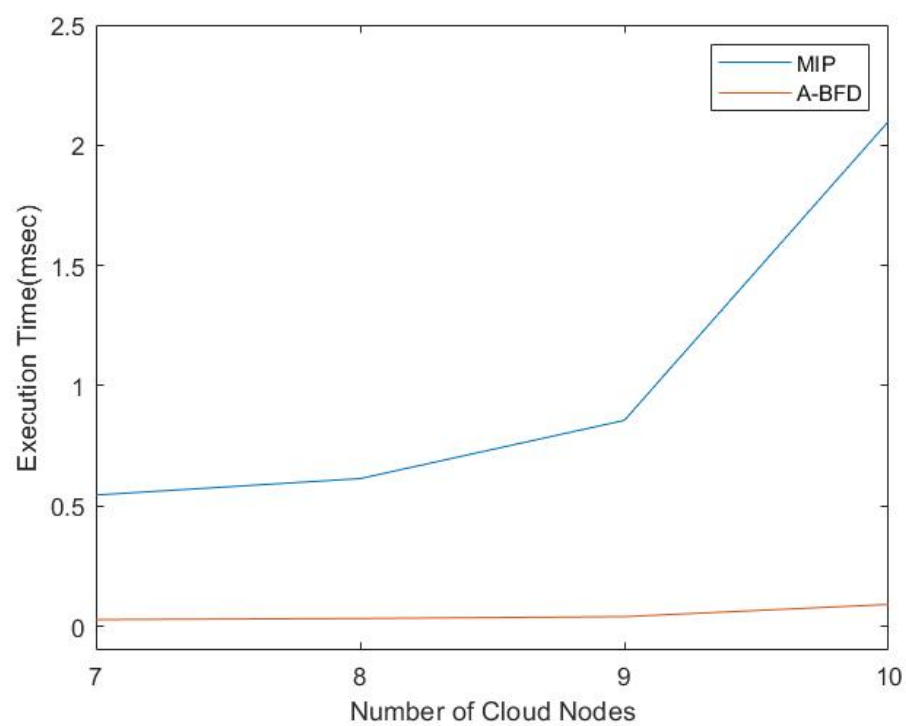
شکل ۴-۱۳: میانگین هزینه تخصیص زنجیره خدمات بر حسب تعداد زنجیره‌های خدمات



شکل ۴-۱۴: میانگین هزینه تخصیص زنجیره خدمات بر حسب اندازه بافر لینک‌ها



شکل ۴-۱۵: میانگین تعداد سرورهای تخصیص یافته



شکل ۴-۱۶: زمان لازم جهت اجرای الگوریتم‌ها

فصل ۵

جمع‌بندی و پیشنهادها

۵-۱ مقدمه

در این فصل خلاصه‌ای از یافته‌های تحقیق جاری خواهد شد. همچنین نتایج به دست آمده جمع‌بندی شده و در نهایت پیشنهادهایی برای کارهای آینده ارائه خواهد شد.

۵-۲ نتیجه‌گیری

ظهور دو فناوری جدید، یعنی شبکه‌های نرم‌افزار محور (SDN) و مجازی‌سازی توابع شبکه، (NFV) به‌طور اساسی به‌کارگیری و معماری شبکه را تغییر داده است. این دو فناوری به اپراتورهای موبایل وعده‌هایی چون کاهش هزینه‌ها، افزایش انعطاف‌پذیری، افزایش مقیاس‌پذیری و کاهش زمان لازم برای عرضه برنامه‌های کاربردی و سرویس‌های جدید به بازار را می‌دهند.

با ظهور SDN و NFV و مزایای ارائه‌شده آن‌ها، اپراتورهای تلفن همراه به‌تدریج نحوه طراحی شبکه‌های تلفن همراه خود را تغییر می‌دهند تا با رشد روزافزون ترافیک داده، تعداد زیاد دستگاه‌های جدید و دسترسی به شبکه همگام شوند و مسیر خود را برای حرکت به‌سوی نسل پنجم شبکه‌ها هموار کنند.

در گزارش ارائه شده، ابتدا مفاهیم اساسی‌ای چون شبکه نرم‌افزار محور، مجازی‌سازی توابع شبکه و مسیریابی

مطرح شدند. سپس مسئله هماهنگ‌سازی یا جاییابی توابع شبکه و مسیریابی با مدیریت SDN مطرح شد. به صورت خاص، دورش اکتشافی برای مواجهه با این مسئله مطرح شده و عملکرد یکی از آن‌ها مورد تحلیل قرار گرفت.

۳-۵ پیشنهادات

با توجه به نتایج حاصل از این پژوهش موضوعات زیر برای پژوهش‌های آتی پیشنهاد می‌گردند.

- بررسی گسترده و دقیق قیود کیفیت سرویسی مختلف جهت دست‌یابی به سرویس با کیفیت مطلوب
- بررسی جامع هماهنگ‌سازی NFV و SDN
- استفاده از مشخصات آماری ورود و خروج زنجیره‌های سرویس به شبکه

- [1] Dijkstra's algorithm, Dec 2019.
- [2] Heuristic (computer science), Dec 2019.
- [3] Np-hardness, Dec 2019.
- [4] Addis, Bernardetta, Belabed, Dallal, Bouet, Mathieu, and Secci, Stefano. Virtual network functions placement and routing optimization. In *2015 IEEE 4th International Conference on Cloud Networking (CloudNet)*, pages 171–177. IEEE, 2015.
- [5] Alves, Cláudio and de Carvalho, José M. Valério. Accelerating column generation for variable sized bin-packing problems. *European Journal of Operational Research*, 183:1333–1352, 2007.
- [6] Belov, Gleb and Scheithauer, Guntram. A cutting plane algorithm for the one-dimensional cutting stock problem with multiple stock lengths. *European Journal of Operational Research*, 141:274–294, 2002.
- [7] Cappanera, Paola, Paganelli, Federica, and Paradiso, Francesca. Vnf placement for service chaining in a distributed cloud environment with multiple stakeholders. *Computer Communications*, 133:24–40, 2019.
- [8] Carugi, Marco. Key features and requirements of 5g/imt-2020 networks. *Internet-society. org, February*, 2018.
- [9] Crainic, Teodor Gabriel, Perboli, Guido, Rei, Walter, and Tadei, Roberto. Efficient heuristics for the variable size bin packing problem with fixed costs. 2010.
- [10] Dräxler, Sevil, Karl, Holger, and Mann, Zoltán Ádám. Jasper: Joint optimization of scaling, placement, and routing of virtual network services. *IEEE Transactions on Network and Service Management*, 15(3):946–960, 2018.

- [11] ETSI, GSNFV. Network functions virtualisation (nfv): Architectural framework. *ETSI Gs NFV*, 2(2):V1, 2013.
- [12] Fredman, Michael L. and Tarjan, Robert E. Fibonacci heaps and their uses in improved network optimization algorithms. In *FOCS*, 1984.
- [13] Friesen, D. K. and Langston, M. A. Variable sized bin packing. *SIAM J. Comput.*, 15:222–230, 1986.
- [14] Gouareb, Racha, Friderikos, Vasilis, and Aghvami, Abdol-Hamid. Virtual network functions routing and placement for edge cloud latency minimization. *IEEE Journal on Selected Areas in Communications*, 36(10):2346–2357, 2018.
- [15] Grossmann, Thomas and Flitter, Helmut.
- [16] Gupta, Abhishek, Habib, M Farhan, Chowdhury, Pulak, Tornatore, Massimo, and Mukherjee, Biswanath. Joint virtual network function placement and routing of traffic in operator networks. *UC Davis, Davis, CA, USA, Tech. Rep*, 2015.
- [17] Gushchin, Andrey, Walid, Anwar, and Tang, Ao. Scalable routing in sdn-enabled networks with consolidated middleboxes. In *Proceedings of the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, pages 55–60. ACM, 2015.
- [18] Haouari, Mohamed and Serairi, Mehdi. Heuristics for the variable sized bin-packing problem. *Computers & OR*, 36:2877–2884, 2009.
- [19] Haouari, Mohamed and Serairi, Mehdi. Relaxations and exact solution of the variable sized bin packing problem. *Computational Optimization and Applications*, 48:345–368, 2011.
- [20] Hirwe, Anish and Kataoka, Kotaro. Lightchain: A lightweight optimisation of vnf placement for service chaining in nfv. In *2016 IEEE NetSoft Conference and Workshops (NetSoft)*, pages 33–37. IEEE, 2016.
- [21] Jang, Insun, Suh, Dongeun, Pack, Sangheon, and Dán, György. Joint optimization of service function placement and flow distribution for service function chaining. *IEEE Journal on Selected Areas in Communications*, 35(11):2532–2541, 2017.

- [22] Kang, Jangha and Park, Sungsoo. Algorithms for the variable sized bin packing problem. *European Journal of Operational Research*, 147:365–372, 2003.
- [23] Khebbache, Selma, Hadji, Makhlouf, and Zeghlache, Djamal. Virtualized network functions chaining and routing algorithms. *Computer Networks*, 114:95–110, 2017.
- [24] KUMAR, SP AJITH. *ROUTING AND CONGESTION CONTROL IN OPPORTUNISTIC NETWORKS*. PhD thesis, 2016.
- [25] Leivadeas, Aris, Falkner, Matthias, Lambadaris, Ioannis, and Kesidis, George. Optimal virtualized network function allocation for an sdn enabled cloud. *Computer Standards & Interfaces*, 54:266–278, 2017.
- [26] Li, Yong and Chen, Min. Software-defined network function virtualization: A survey. *IEEE Access*, 3:2542–2553, 2015.
- [27] 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, 2017.
- [28] Mijumbi, Rashid, Serrat, Joan, and Gorricho, Juan-Luis. Self-managed resources in network virtualisation environments. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 1099–1106. IEEE, 2015.
- [29] Mohammadkhan, Ali, Ghapani, Sheida, Liu, Guyue, Zhang, Wei, Ramakrishnan, KK, and Wood, Timothy. Virtual function placement and traffic steering in flexible and dynamic software defined networks. In *The 21st IEEE International Workshop on Local and Metropolitan Area Networks*, pages 1–6. IEEE, 2015.
- [30] Monaci, Michele. Algorithms for packing and scheduling problems. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 1:85–87, 2003.
- [31] Pisinger, David and Sigurd, Mikkel. The two-dimensional bin packing problem with variable bin sizes and costs. *Discrete Optimization*, 2:154–167, 2005.
- [32] Qazi, Zafar Ayyub, Tu, Cheng-Chun, Chiang, Luis, Miao, Rui, Sekar, Vyas, and Yu, Minlan. Simple-fying middlebox policy enforcement using sdn. In *ACM SIGCOMM computer communication review*, volume 43, pages 27–38. ACM, 2013.

- [33] Qu, Long, Assi, Chadi, Shaban, Khaled, and Khabbaz, Maurice J. A reliability-aware network service chain provisioning with delay guarantees in nfv-enabled enterprise datacenter networks. *IEEE Transactions on Network and Service Management*, 14(3):554–568, 2017.
- [34] Rawat, Danda B and Reddy, Swetha R. Software defined networking architecture, security and energy efficiency: A survey. *IEEE Communications Surveys & Tutorials*, 19(1):325–346, 2016.
- [35] Rieck, Bastian. Basic analysis of bin-packing heuristics. 2009.
- [36] Rothlauf, Franz. Design of modern heuristics: Principles and application. 2011.
- [37] Tuysuz, Mehmet Fatih, Ankarali, Zekiye Kubra, and Gözüpek, Didem. A survey on energy efficiency in software defined networks. *Computer Networks*, 113:188–204, 2017.
- [38] Zhang, Hailong and Yan, Jinyao. Performance of sdn routing in comparison with legacy routing protocols. In *2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pages 491–494. IEEE, 2015.
- [39] Zhang, Ying, Beheshti, Neda, Beliveau, Ludovic, Lefebvre, Geoffrey, Manghir-malani, Ravi, Mishra, Ramesh, Patneyt, Ritun, Shirazipour, Meral, Subrahmaniam, Ramesh, Truchan, Catherine, et al. Steering: A software-defined networking for inline service chaining. In *2013 21st IEEE international conference on network protocols (ICNP)*, pages 1–10. IEEE, 2013.

پیوست آ

روش مجموع زیرمجموعه

مساله مجموع زیرمجموعه^۱ یا SSP، یک مساله مهم در علوم کامپیوتر است و هدف آن یافتن زیرمجموعه‌ای از یک مجموعه داده شده به گونه‌ای است که مجموع اجزا آن زیرمجموعه برابر با یک عدد معین باشد. طبق برخی نتایج تجربی، در صورتی که در پاسخ بهینه BPP، هر مکان در صورت استفاده شدن تقریباً به صورت کامل پر شود (یعنی در حالتی که اشیاء دارای وزن کم باشند)، حل به روش پاسخ به SSP برای حل BPP به روش‌های اکتشافی کلاسیک همچون روش‌های اولین برآزش کاهشی و بهترین برآزش کاهشی ارجحیت خواهد داشت.

SSP، NP کامل است، یعنی با وجود این که تشخیص این که آیا یک پاسخ ارائه شده صحیح می‌باشد یا خیر، راحت است اما ممکن است ذاتاً تعیین این که آیا در وهله اول اصلاً پاسخی وجود دارد یا خیر به طرز بازدارنده‌ای دشوار باشد. این مساله همچنین حالت خاصی از مساله کوله‌پشتی ۱-۰ نیز می‌باشد و نتیجتاً می‌توان از روش‌های حل مساله کوله‌پشتی برای آن استفاده کرد. هرچند، الگوریتم‌هایی که به طور خاص برای یک مساله طراحی شده باشند معمولاً نتیجه بهتری در بر خواهند داشت.

چندین روش برای حل SSP در زمان نمایی وجود دارد. ساده‌ترین روش، روش جستجوی جامع می‌باشد و به این صورت است که ابتدا مجموعه تمام زیرمجموعه‌های ممکن (مجموعه توانی) مجموعه داده شده را تولید شده و سپس بررسی می‌گردد که آیا مجموع اجزا هیچ یک از این زیرمجموعه‌ها برابر با عدد خواسته شده خواهد بود یا خیر. زمان اجرای این روش از مرتبه $O(2^n)$ خواهد بود.

از جمله دیگر روش‌هایی که می‌توانند برای حل مساله SS مورد استفاده قرار گیرند به صورت زیر هستند.

¹subset sum problem

- روش بازگشتی

- روش برنامه نویسی دینامیکی

- روش عقبگرد

در ادامه روش عقبگرد مختصراً شرح داده خواهد شد.

آ-۰-۱ روش عقبگرد

در روش جستجوی جامع، تمام زیر مجموعه‌ها بدون توجه به اینکه آیا قید خواسته شده را برقرار می‌کنند یا خیر، در نظر گرفته می‌شوند اما روش عقبگرد^۲ می‌تواند جهت اعمال توجه سازماندهی شده به انتخاب المان‌ها مورد استفاده قرار گیرد. به طور خلاصه، گام‌های این روش به صورت زیر خواهند بود.

۱. با یک زیر مجموعه خالی شروع می‌شود.

۲. یک جزء از لیست به زیر مجموعه اضافه می‌شود.

۳. اگر مجموع اجزای زیر مجموعه برابر با عدد خواسته شده باشد، روند متوقف شده و آن زیر مجموعه به عنوان پاسخ ارائه می‌گردد.

۴. اگر زیر مجموعه شدنی نباشد (یعنی مجموع اجزای زیر مجموعه بیشتر از عدد خواسته شده باشد) یا زیر مجموعه به آخر رسیده باشد، از میان زیر مجموعه عقبگرد صورت می‌گیرد تا هنگامی که مناسب‌ترین مقدار پیدا شود.

۵. اگر زیر مجموعه شدنی باشد (یعنی مجموع اجزای زیر مجموعه کمتر از عدد خواسته شده باشد)، روند از مرحله ۲ تکرار می‌شود.

۶. اگر تمام اجزای بررسی شدند و زیر مجموعه مناسبی یافت نشد و یا اگر عقبگرد ممکن نبود، آن‌گاه بدون یافتن پاسخ، روند متوقف می‌گردد.

²backtracking

در روش عقب‌گرد، در حین پیشروی در عمق درختی که تا کنون اجزا به آن اضافه شده‌اند، اگر جزء اضافه شده قید مجموع را برآورده کند، ایجاد زیرگره‌ها ادامه پیدا خواهد داد. هر زمان که قید مجموع برآورده نشود، ایجاد شاخه‌های اضافی در آن گره متوقف می‌شود و برای کشف گره‌هایی که هنوز مورد کاوش قرار نگرفته‌اند، به گره قبلی عقب‌گرد صورت می‌گیرد. می‌بایست گره‌ها در طول و عرض درخت مورد کاوش قرار گیرند. شبه کد این روش به صورت آ-۱ است.

Algorithm Backtracking

- 1: if subset is satisfying the constraint
 - 2: print the subset
 - 3: exclude the current element and consider next element
 - 4: else
 - 5: generate the nodes of present level along breadth of tree and recurse
 - 6: for next levels
-

شکل آ-۱: شبه کد الگوریتم عقب‌گرد

پیوست ب

ارزیابی عملکرد

عملکرد یک الگوریتم تقریبی را می توان با بررسی حدود عملکرد بدترین حالت اندازه گرفت. دو نوع حد قابل تعریف است:

- حد عملکرد بدترین حالت تقریبی
- حد عملکرد بدترین حالت مطلق

اگر برای یک مورد مشخص I ، $C_H(I)$ و $C^*(I)$ به ترتیب مقدار به دست آمده توسط الگوریتم تقریبی H و مقدار بهینه باشند، تحلیل بدترین حالت تقریبی شامل یافتن رابطه به فرم رابطه ب-۱ است که در آن A و R ثابت های غیر منفی مستقل از I هستند.

$$C_H(I) \leq R * C^*(I) + A \quad \forall I \quad (\text{ب-۱})$$

همچنین، تحلیل بدترین حالت مطلق، رابطه ای به فرم ب-۲ برقرار می کند.

$$C_H(I) \leq \Omega * C^*(I) \quad \forall I \quad (\text{ب-۲})$$

کوچکترین مقادیر R و Ω که روابط ب-۱ و ب-۲ را برقرار می کنند، به ترتیب، نسبت عملکرد بدترین حالت تقریبی و نسبت عملکرد بدترین حالت مطلق نامیده می شوند. این حدود به طور گسترده جهت ارزیابی عملکرد

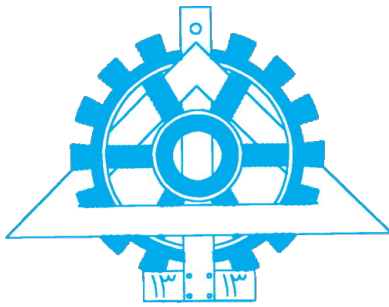
روش های ارائه شده در پژوهش ها به کار می روند.

Abstract:

The fifth generation of wireless networks has been proposed as a platform to provide considerably higher network capacity, enable massive device connectivity with reduced latency and cost, and achieve considerable energy savings. In this regard, network design, management and upgrades require the use of innovative technologies and architectures. Enterprise and service provider networks are increasingly making use of Virtualized Network Functions (VNFs) and Software-Defined Networking (SDN) to reap the benefits of reduced Capital expenditures (CAPEX) and Operating expenses (OPEX).

In this research, the problem of placing virtual network functions on existing resources as well as data routing through these functions to execute a particular service chain is investigated. A network consisting of switches, servers and links with determined specifications is considered and an optimization problem is designed to minimize the total cost of links and servers utilization, considering different constraints. These constraints include network constraints, topology constraints and the constraint on maximum congestion of links. Here, in order to estimate costs, both the service provider view and the cloud provider requirements are taken into account. In the following, two heuristic algorithms to solve the optimization problem are presented and the performance of one of them is evaluated by simulation.

Keywords: 5G, Software-Defined Networking, Network Function Virtualization, Routing, Heuristic



University of Tehran
College of Engineering
School of Electrical and Computer
Engineering
Communication



Orchestration of Routing and VNF Placement in 5G Networks

A Thesis submitted to the Graduate Studies Office
In partial fulfillment of the requirements for
The degree of Master of Science
in 5G networks - Telecommunication Networks

By:

Gelareh Haselmehri

Supervisor:

Dr. Vahid Shah-mansouri

September 2019