

# جایابی چند هدفه و مبتنی بر یادگیری ماشین کنترل کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار

مبین قلی‌زاده و ناصر مزینی

شبکه‌های کامپیوتری مطرح شده است. این شبکه‌ها با جداسازی بخش کنترلی و بخش داده و استفاده از کنترل کننده مرکزی به اهداف مورد نظر خود از جمله عدم وابستگی به نوع سخت‌افزار، کاهش پیچیدگی، افزایش انعطاف‌پذیری و سرعت، افزایش کارایی و کاهش هزینه راه‌اندازی و مدیریت شبکه‌ها دست می‌یابند. همچنین با استفاده از معماری SDN به مزایایی همچون بهبود کارآمدی، بهره‌وری، قابلیت اطمینان، مدیریت خودکار، توسعه‌پذیری و تأمین امنیت در شبکه‌های کامپیوتری می‌توان دست یافت [۱] تا [۳].

از آنجایی که برای مدیریت جریان ترافیک، سوئیچ‌ها و میزبان‌های شبکه SDN، وجود یک کنترل کننده به منظور پاسخگویی به میزان تقاضای روزافزون شبکه کافی نیست، نیاز به استقرار کنترل کننده‌های متعددی وجود دارد. همچنین، کنترل کننده متمرکز و واحد به تنها نقطه شکست در شبکه تبدیل شده و این امر سبب افزایش آسیب‌پذیری شبکه در برابر حملات، اختلالات و ناپایداری شبکه می‌شود. علاوه بر این شبکه مبتنی بر کنترل کننده متمرکز ممکن است از چالش‌های دیگری از جمله مقیاس‌پذیری، سربار ترافیکی در محدوده کنترل کننده، نیاز به توان پردازشی بالا، قابلیت ذخیره‌سازی حجم زیادی از داده‌ها و گذردهی مناسب برای پاسخ به ترافیک‌های ارسالی رنج می‌برد. برای مقابله با این چالش‌ها معماری توزیع شده برای بخش کنترلی با حفظ دید متمرکز پیشنهاد شده است از آنجایی که برای مدیریت جریان ترافیک، سوئیچ‌ها و میزبان‌های شبکه SDN، وجود یک کنترل کننده به منظور پاسخگویی به میزان تقاضای روزافزون شبکه کافی نیست، نیاز به استقرار کنترل کننده‌های متعددی وجود دارد. همچنین، کنترل کننده متمرکز و واحد به تنها نقطه شکست در شبکه تبدیل شده و این امر سبب افزایش آسیب‌پذیری شبکه در برابر حملات، اختلالات و ناپایداری شبکه می‌شود. علاوه بر این شبکه مبتنی بر کنترل کننده متمرکز ممکن است از چالش‌های دیگری از جمله مقیاس‌پذیری، سربار ترافیکی در محدوده کنترل کننده، نیاز به توان پردازشی بالا، قابلیت ذخیره‌سازی حجم زیادی از داده‌ها و گذردهی مناسب برای پاسخ به ترافیک‌های ارسالی رنج می‌برد. برای مقابله با این چالش‌ها معماری توزیع شده برای بخش کنترلی با حفظ دید متمرکز پیشنهاد شده است [۴] و [۵].

در معماری SDN با چندین کنترل کننده، مسئله اصلی تعیین تعداد و مکان استقرار بهینه کنترل کننده‌ها در شبکه می‌باشد. با وجود پژوهش‌های صورت گرفته برای جایابی و استقرار کنترل کننده‌ها در SDN، به لحاظ فاکتورهای کیفیت خدمات در مدل‌سازی جایابی کنترل کننده‌ها توجه کافی نشده است. در صورتی که لحاظ پارامترهای کیفیت خدمات همچون توان عملیاتی، تأخیر، پایداری، تعادل بار و اطمینان‌پذیری از اهمیت بالایی در این شبکه‌ها برخوردار بوده و همچنین ارائه عملکرد در این شبکه‌ها را

چکیده: امروزه با رشد روزافزون شبکه‌های کامپیوتری و افزایش پیچیدگی ساختارهای ارتباطی، نیاز به مدیریت بهینه منابع، کاهش هزینه‌های عملیاتی و نگهداری به چالشی اساسی تبدیل شده است. شبکه‌های مبتنی بر نرم‌افزار با جداسازی لایه داده و کنترل، انعطاف‌پذیری، مقیاس‌پذیری و مدیریت متمرکز را فراهم آورده است. در لایه کنترل، کنترل کننده‌ها به عنوان عنصر اصلی مدیریت و تصمیم‌گیری در شبکه عمل کرده و چالش اصلی در این حوزه جایابی بهینه آن‌ها در شبکه است. نحوه توزیع و استقرار کنترل کننده‌ها تأثیر مستقیمی بر عملکرد شبکه همچون کاهش تأخیر، برقراری تعادل بار و افزایش پایداری شبکه دارد. در این مقاله، به ارائه یک روش چندهدفه مبتنی بر یادگیری ماشین برای جایابی کنترل کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار پرداخته شده است. به این منظور، مسئله را با هدف کینه‌کردن تأخیر لایه کنترل و تأخیر میان‌دامنه‌ای، بار لایه کنترل و هزینه شکست گره‌ها و پیوندهای توپولوژی شبکه و همچنین پیشینه‌کردن توان عملیاتی مدل‌سازی شده است. همچنین، با توجه به پیچیدگی محاسباتی مسئله، یک الگوریتم حل مبتنی بر یادگیری تقویتی عمیق مورد استفاده قرار گرفته است. نتایج شبیه‌سازی‌ها در چند توپولوژی مختلف نشان می‌دهد که الگوریتم پیشنهادی توانسته است تأخیر لایه کنترل، تأخیر درون دامنه‌ای و توان عملیاتی را به ترتیب تا ۲۰٪، ۳۰٪ و ۱۵ درصد نسبت به کارهای پیشین بهبود بخشد. بدین ترتیب، رویکرد پیشنهادی علاوه بر پوشش اهداف متداول همچون کاهش تأخیر و توازن بار، معیارهای مهمی مانند پایداری و تحمل‌پذیری شکست را نیز در فرآیند تصمیم‌گیری جایابی در نظر می‌گیرد.

کلیدواژه: جایابی کنترل کننده‌ها، شبکه‌های مبتنی بر نرم‌افزار، یادگیری ماشین.

## ۱- مقدمه

در دنیای امروز، همزمان با رشد سریع شبکه‌های کامپیوتری، افزایش تعداد سوئیچ‌ها و ترافیک موجود، معماری فعلی شبکه‌ها با مشکلات و چالش‌های اساسی مواجه است. از جمله این چالش‌ها، هزینه بالای راه‌اندازی و مدیریت تعداد تجهیزات زیاد و پیوند میان آن‌ها، تأمین امنیت و عیب‌یابی این تجهیزات، ارائه عملکرد بهینه و سرعت مناسب برای برنامه‌های کاربردی جدید است. در سال‌های اخیر شبکه‌های مبتنی بر نرم‌افزار<sup>۱</sup> (SDN) برای مدیریت ساده‌تر، بهینه و قابلیت برنامه‌ریزی بیشتر

این مقاله در تاریخ ۲۵ فروردین ماه ۱۴۰۴ دریافت و در تاریخ ۱۶ آبان ماه ۱۴۰۴ بازنگری شد.

مبین قلی‌زاده، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران، (email: mobin\_gholizadeh@comp.iust.ac.ir).

ناصر مزینی (نویسنده مسئول)، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران، (email: mozayani@iust.ac.ir).

و هدف، فضای حالت مسئله با ابعاد بالا را در زمان خوبی جستجو کرده و تصمیم مناسب در جهت اهداف مسئله اخذ می‌نماید.

- در نهایت، به منظور ارزیابی عملکرد، روش پیشنهادی و یکی از کارهای مرتبط را پیاده‌سازی نموده‌ایم. نتایج حاصل از شبیه‌سازی‌های گسترده تحت سناریوهای مختلف نشان می‌دهد روش پیشنهادی معیارهای هدف به ویژه تأخیر لایه کنترل، تأخیر میان‌دامنه‌ای و توان عملیاتی را به طور قابل توجهی به نسبت کار مرتبط بهبود می‌بخشد. به علاوه از اهداف بیشتری به نسبت کار مورد مقایسه پشتیبانی می‌کند.

ساختار ادامه این مقاله به شرح زیر است: در بخش دوم، به بررسی کارهای انجام‌شده در زمینه روش‌های چندهدفه جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار می‌پردازیم. در بخش سوم، ضمن ارائه معماری پیشنهادی، معیارهای تشکیل‌دهنده تابع هدف را مدل‌سازی می‌نماییم و مسئله‌ی چند هدفه جایابی کنترل‌کننده‌ها را فرموله‌سازی می‌کنیم. در بخش چهارم، الگوریتم مورد استفاده برای حل مسئله مدل‌سازی شده ارائه گردیده است. در بخش پنجم، نتایج ارزیابی و مقایسه روش پیشنهادی با روش مشابه عنوان شده است. در نهایت، در بخش آخر نتیجه‌گیری و پیشنهادهایی برای کارهای آتی مطرح می‌نماییم.

## ۲- مروری بر کارهای پیشنهادی

در این بخش به منظور آشنایی با روش‌های چندهدفه جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار، به بررسی کارهای مرتبط اخیر موجود در این حوزه پرداخته می‌شود.

در کار [۹]، هدف از جایابی کنترل‌کننده‌ها افزایش قابلیت اطمینان و برقراری تعادل بار در شبکه است. در این پژوهش قابلیت اطمینان برای دو پیوند یکی میان سوئیچ‌ها و کنترل‌کننده مربوط به آن‌ها و دیگری میان دو کنترل‌کننده، تعریف می‌گردد. همچنین، فرض بر این است که شکست در گره‌ای از توپولوژی رخ نمی‌دهد و تنها در پیوندها قابل به وقوع پیوستن است. این مسئله با رویکرد اکتشافی مبتنی بر الگوریتم ژنتیک مورد حل قرار گرفته است.

در کار [۱۰] هدف، جایابی کنترل‌کننده‌ها براساس کاهش تأخیر و برقراری توازن بار است. همچنین، این پژوهش تعداد کنترل‌کننده‌ها را نیز برای توپولوژی تعیین می‌کند. منظور از تأخیر در این مسئله تأخیر میان سوئیچ‌ها و کنترل‌کننده مربوط به خود و همچنین تأخیر میان دو کنترل‌کننده است. به منظور برقراری توازن بار نیز معیار عدم توازن در کنترل‌کننده‌ها تعریف گردیده که می‌بایست کمینه گردد. این مسئله با سه رویکرد فرااکتشافی، اکتشافی و الهام‌گرفته از طبیعت مورد حل قرار گرفته است. این پژوهش از بررسی رویکردهای مبتنی بر یادگیری ماشین و یادگیری عمیق غافل مانده است.

در کار [۱۱] هدف، جایابی کنترل‌کننده‌ها براساس کمینه‌کردن میانگین تأخیر و بیشینه‌کردن میانگین تعداد مسیر است. منظور از تأخیر در این پژوهش، تأخیر میان دو کنترل‌کننده و تیخیر میان سوئیچ و کنترل‌کننده است. از آنجایی که وجود تنها یک مسیر میان سوئیچ و کنترل‌کننده تخصیص‌یافته به آن می‌تواند شبکه را با شکست مواجه کند، افزایش تعداد مسیرها میان سوئیچ و کنترل‌کننده نیز در جایابی کنترل‌کننده‌ها موثر است. این کار با الگوریتم بهینه‌سازی الهام‌گرفته از طبیعت، شیر مورچه مورد حل قرار گرفته است.

در کار [۱۲] هدف از جایابی کنترل‌کننده‌ها کمینه‌کردن میانگین تأخیر شبکه و معیار انحراف تعادل بار است. میانگین تأخیر از تأخیر انتشار، تأخیر

وابسته به نوع خدمات خواهد کرد. همچنین، کارهای مرتبط به مسئله لحاظ اهداف ترکیبی در جایابی کنترل‌کننده‌ها توجه نکرده‌اند یا در صورت توجه غالباً به اهدافی یکسان و محدود شامل کاهش تأخیر و برقراری تعادل بار پرداخته‌اند و توجه به معیارهای مهمی همچون افزایش انعطاف‌پذیری، پایداری، ثبات و توان عملیاتی در شبکه مغفول واقع شده است. این در حالی است که در جایابی کنترل‌کننده‌ها بایستی اهداف مختلفی را مورد نظر قرار داد تا به کارآمدی در شبکه دست یافت. همچنین، در غالب کارهای مرتبط از رویکردهای اکتشافی و فرااکتشافی برای حل مسئله جایابی کنترل‌کننده‌ها استفاده شده است. این در صورتی است که، رویکردهای مبتنی بر یادگیری ماشین بخصوص یادگیری تقویتی به علت توانایی بالا در انطباق با تغییرات پویای شبکه و بهبود معیارها برای کاربردهای جدید و بلادرنگ به اخذ تصمیم مناسب‌تر برای جایابی کنترل‌کننده‌ها در شرایط پویا و با لحاظ چند معیار دست می‌یابند [۶] و [۷].

برای پرداختن به این چالش‌ها، در این مقاله به ارائه یک روش چندهدفه مبتنی بر یادگیری ماشین برای جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار پرداخته شده است. به این منظور، در این مقاله مسئله جایابی کنترل‌کننده‌ها را با هدف کمینه‌کردن تأخیر لایه کنترل، میانگین بار لایه کنترل، تأخیر میان‌دامنه‌ای، هزینه شکست گره‌ها و پیوندهای توپولوژی شبکه و همچنین بیشینه‌کردن توان عملیاتی مدل‌سازی نموده‌ایم. در این راستا، قیود عدم همپوشانی در دامنه‌های تشکیل‌دهنده لایه داده، نگاشت سوئیچ‌ها به نزدیک‌ترین کنترل‌کننده و تخصیص تنها یک کنترل‌کننده به یک دامنه نیز مورد توجه قرار گرفته است. همچنین، به منظور حل مسئله مدل‌سازی شده، از یک الگوریتم مبتنی بر یادگیری تقویتی عمیق استفاده می‌کنیم. در این راستا، حالت‌ها، اقدامات و تابع پاداش برای عامل یادگیری تقویتی عمیق با توجه به فضای مسئله طراحی شده است. این مقاله با لحاظ اهداف متعدد در جایابی کنترل‌کننده‌ها و همچنین استفاده از قابلیت‌های یادگیری ماشین بخصوص یادگیری تقویتی به بهبود در معیارهای ارزیابی همچون کاهش تأخیر و افزایش پایداری و ثبات دست یافته است. اگرچه در پژوهش‌های پیشین همچون مرجع [۸] مسئله جانمایی کنترل‌کننده‌ها با رویکرد چندهدفه و مبتنی بر یادگیری تقویتی بررسی شده است، اما در این مقاله ترکیب خاصی از اهداف شامل پایداری گره و پیوند، تأخیر کنترل و میان‌دامنه‌ای، توان عملیاتی و تعادل بار به‌طور هم‌زمان در مدل‌سازی تابع هدف مد نظر قرار گرفته که در ادبیات پیشین مغفول مانده است. به طور خلاصه، نوآوری‌های این مقاله به شرح زیر است:

- با توجه به اینکه در کارهای قبلی غالباً دو معیار تأخیر و برقراری تعادل بار برای جایابی کنترل‌کننده‌ها مد نظر قرار می‌گرفته است؛ در این مقاله معیار پایداری و ثبات به مسئله اضافه گشته که با تشریح مفهوم درجه مرکزیت گره‌ها و پیوندها در توپولوژی شبکه به کاهش هزینه شکست در شبکه و افزایش پایداری منجر می‌گردد. در واقع، برای نخستین بار، معیار پایداری گره‌ها و پیوندها (براساس درجه مرکزیت) به صورت صریح در تابع هدف مدل‌سازی شده و به شکل هم‌زمان با سایر معیارها بهینه‌سازی شده است.
- با توجه به NP-hard بودن مسئله مدل‌سازی شده و عدم امکان حل آن در مدت زمان قابل قبول، در ادامه یک الگوریتم مبتنی بر یادگیری تقویتی عمیق برای حل این مسئله با چند هدف مورد استفاده قرار می‌دهیم. الگوریتم حل مورد استفاده با دریافت بازخورد از محیط با نرخ پویایی بالا تطبیق داشته و با بهره‌گیری از شبکه‌های عصبی پایه

تخصیص کنترل‌کننده‌ها می‌نماید. نتایج نشان می‌دهد روش پیشنهادی نسبت به رویکردهای سنتی از نظر توازن بار و کارایی شبکه عملکرد بهتری دارد.

در کار [۱۸]، مسئله جایابی کنترل‌کننده‌ها در محیط SDN-IoV با در نظر گرفتن کیفیت خدمات و بار شبکه بررسی شده است. در این کار از الگوریتم یادگیری تقویتی عمیق DDPG استفاده شده تا جایابی کنترل‌کننده‌ها به صورت پویا و پیوسته با تغییر شرایط ترافیکی انجام گیرد. نتایج شبیه‌سازی نشان داده‌اند که این روش موجب کاهش تأخیر انتها به انتها و بهبود QoS در سناریوهای متحرک می‌شود.

در کار [۱۹] یک رویکرد یادگیری تقویتی برخط برای برقراری توازن بار میان کنترل‌کننده‌های SDN پیشنهاد شده است. عامل یادگیرنده با مشاهده‌ی بار لحظه‌ای کنترل‌کننده‌ها و تأخیر پاسخ، سیاست تغییر تخصیص سوئیچ‌ها را به گونه‌ای تنظیم می‌کند که بار میان کنترل‌کننده‌ها همگن گردد. این روش بدون نیاز به داده‌های آموزشی از پیش تعیین شده، به صورت تطبیقی با تغییرات ترافیکی عمل کرده و به کاهش تأخیر در لایه کنترل منجر می‌شود.

در کار [۲۰] مسئله‌ی جایابی کنترل‌کننده‌ها و مسیریابی در SDN با در نظر گرفتن دو معیار قابلیت اطمینان و ظرفیت کنترل‌کننده‌ها بررسی شده است. در این پژوهش، یک چارچوب ترکیبی مبتنی بر هوش مصنوعی شامل الگوریتم ژنتیک و شبکه عصبی برای انتخاب جایگاه کنترل‌کننده‌ها و مسیرهای مرتبط طراحی گردیده است. هدف مدل، پیشینه‌سازی قابلیت اطمینان و بهره‌وری ظرفیت در کنار کاهش تأخیر است.

با بررسی مطالعات پیشین، نیازمندی‌های اصلی برای طراحی یک روش جانمایی کنترل‌کننده‌ها در SDN، کمینه‌سازی تأخیر، برقراری تعادل بار در توپولوژی و توان عملیاتی بالا استخراج شده است. بررسی کارهای مرتبط مشخص می‌کند که با وجود تحقیقات در زمینه روش‌های چندهدفه جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار، این کارها غالباً به اهدافی یکسان شامل کاهش تأخیر و برقراری تعادل بار پرداخته‌اند و توجه به معیارهای مهمی همچون افزایش انعطاف‌پذیری، پایداری، ثبات و توان عملیاتی در شبکه مغفول واقع شده است. همچنین، در غالب کارهای مرتبط از رویکردهای اکتشافی و فرااکتشافی برای حل مسئله جایابی کنترل‌کننده‌ها استفاده شده است. این در حالی است که، رویکردهای مبتنی بر یادگیری ماشین بخصوص یادگیری تقویتی به علت توانایی بالا در انطباق با تغییرات پویای شبکه و بهبود معیارها برای کاربردهای جدید و بالادرنگ به اخذ تصمیم مناسب‌تر برای جایابی کنترل‌کننده‌ها در شرایط پویا و با لحاظ چند معیار دست می‌یابند.

### ۳- مدل سیستم پیشنهادی

به منظور برطرف کردن چالش‌های اشاره‌شده در بخش پیشین، در ادامه به ارائه یک روش چند هدفه مبتنی بر یادگیری ماشین برای جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار می‌پردازیم. به این منظور، در این بخش ابتدا، معماری و اجزای تشکیل‌دهنده سیستم را تشریح کرده و سپس به نحوه مدل‌سازی هدف مسئله یعنی کمینه‌کردن معیارهای کارآمدی شامل بار لایه کنترل، تأخیر لایه کنترل، تأخیر میان‌دامنه‌ای، توان عملیاتی و پایداری گره و پیوند می‌پردازیم. پس از آن در بخش بعد، به ارائه الگوریتم جایابی کنترل‌کننده‌ها مبتنی بر یادگیری Q عمیق پرداخته و با استفاده از آن تخصیص هوشمندانه کنترل‌کننده‌ها به سوئیچ‌ها را رقم می‌زنیم.

پردازش و تأخیر ارسال تشکیل گردیده است. تأخیر انتشار به تعداد گام‌های میان سوئیچ تا کنترل‌کننده وابسته است. تأخیر پردازش وابسته به تعداد سوئیچ‌هایی است که یک کنترل‌کننده به آن‌ها تخصیص یافته است. تأخیر ارسال نیز ناظر به تأخیری است که هر بسته در مسیر سوئیچ تا کنترل‌کننده متحمل می‌شود. این مسئله با برنامه‌ریزی خطی<sup>۱</sup> (LP) فرموله شده و با رویکردی مبتنی بر یادگیری تقویتی مورد حل قرار گرفته است.

در کار [۱۳] هدف از جایابی کنترل‌کننده‌ها، کمینه کردن تأخیر انتشار کلی و افزایش تحمل خطا در شبکه است. بدین منظور این کار با ارائه یک رویکرد اکتشافی، توپولوژی شبکه را به تعدادی زیرشبکه تقسیم می‌کند. معیار تقسیم به صورتی است که سوئیچ‌های داخلی هر زیر شبکه ارتباط نزدیکی با یکدیگر داشته و همچنین حداقل فاصله را با کنترل‌کننده داشته باشند.

در کار [۱۴]، توپولوژی شبکه به تعدادی خوشه براساس یادگیری ماشین مبتنی بر کوانتوم تبدیل می‌شود. رویکرد رایج برای خوشه‌بندی شبکه استفاده از K-means بوده که یکی از روش‌های یادگیری با نظارت است. این پژوهش از Q-means به این منظور استفاده می‌کند و از مفهوم Qbit محاسبات کوانتومی در آن بهره می‌برد. هدف از جایابی کنترل‌کننده‌ها برای هر خوشه کمینه‌کردن تأخیر میان سوئیچ و کنترل‌کننده‌ی تخصیص داده‌شده به آن در فضای خوشه است.

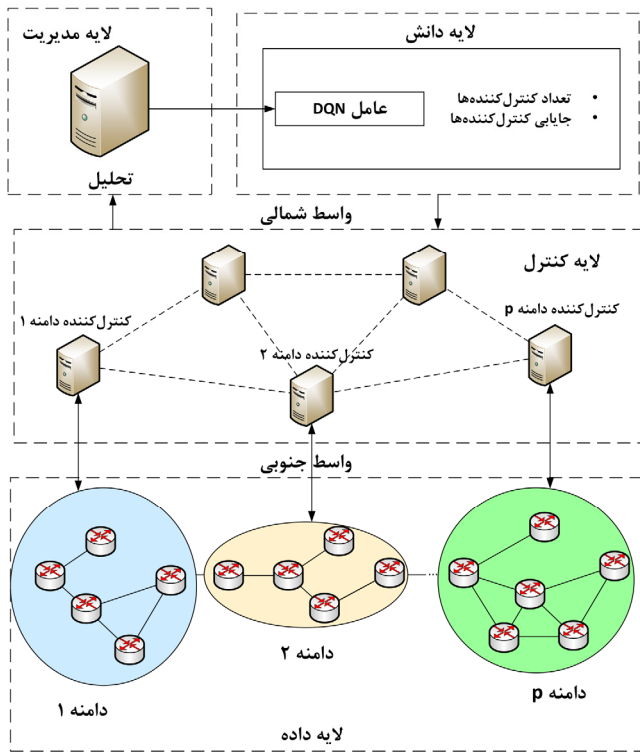
کار [۱۵] به مسئله جایابی کنترل‌کننده‌ها در شبکه‌های نوری انتقال می‌پردازد. از آنجایی که شبکه‌هایی نوری دارای ادوات ناهمگن هستند و پیچیدگی بالایی دارند از معماری سلسله‌مراتبی برای کنترل آن استفاده می‌شود. لایه داده به تعدادی دامنه تبدیل شده که هر دامنه با یک کنترل‌کننده دامنه در ارتباط است. کنترل‌کننده دامنه نیز با کنترل‌کننده والد در لایه بالاتر در ارتباط است. جایابی کنترل‌کننده‌ها تنها در لایه دامنه در این کار مورد توجه است. هدف از جایابی نیز کمینه‌کردن هزینه شامل تأخیر میان سوئیچ و کنترل‌کننده متشکل از تأخیر انتقال، تأخیر انتشار و تأخیر پردازش است. این کار از رویکردی مبتنی بر یادگیری ماشین برای پیش‌بینی تعداد کنترل‌کننده‌ها و جایابی آن‌ها در طول زمان استفاده می‌کند.

در کار [۱۶] اتکا به تعیین مسیر میان کنترل‌کننده و سوئیچ‌های تخصیص یافته به آن بوده است. با تعیین بهینه مسیرها، کنترل‌کننده‌ها جایابی می‌شوند. بدین منظور، معیار توازن بار و تأخیر مورد توجه هستند. این مسئله با یک رویکرد مبتنی بر یادگیری تقویتی الهام گرفته از زیست مورد حل قرار گرفته است.

کار [۸] هدف از جایابی کنترل‌کننده‌ها را کمینه کردن تأخیر میان کنترل‌کننده‌ها، تأخیر میان یک دامنه از سوئیچ‌ها و معیار تعادل بار می‌داند. همچنین این کار، توان عملیاتی را نیز همزمان پیشینه می‌کند. منظور از توان عملیاتی مقدار داده منتقل شده توسط یک دامنه در یک برش زمانی مشخص است. این مسئله توسط یک رویکرد مبتنی بر یادگیری تقویتی عمیق مورد حل قرار گرفته است.

در کار [۱۷] مسئله جایابی کنترل‌کننده‌ها در SDN به صورت پویا و با هدف برقراری توازن بار و کاهش هزینه‌های ارتباطی با استفاده از یادگیری تقویتی عمیق انجام شده است. در این پژوهش از معماری Actor-Critic برای یادگیری سیاست جایابی به صورت خودکار بهره گرفته شده و مدل با توجه به وضعیت فعلی شبکه (شامل بار ترافیکی و هزینه ارتباطی) اقدام به

### ۳-۱ معماری سیستم



شکل ۱: معماری سیستم.

حال، به منظور پیش‌گیری از عدم توازن بار در برخی کنترل‌کننده‌ها، بار ترافیکی میان مجموعه کنترل‌کننده‌ها مورد تعادل قرار گرفته و با عنوان بار لایه کنترل به صورت زیر محاسبه می‌گردد

$$CL^t = \frac{1}{p} \sum_{i=1}^p |CL_i^t - CL_{avg}^t| \quad (2)$$

متغیر  $CL_{avg}^t$  به معنی میانگین بار برای تمامی کنترل‌کننده‌های برگزیده شده است.

#### ۳-۲-۲ تأخیر لایه کنترل

میانگین زمان پاسخ کنترل‌کننده  $C_i$  در زمان  $t$  به صورت زیر تعریف می‌گردد

$$CD_i^t = PD_i^t + 2 \cdot CMD_i^t \quad (3)$$

در (۳)،  $PD_i^t$  و  $CMD_i^t$  به ترتیب تأخیر پردازش و تأخیر ارتباطی کنترل‌کننده  $C_i$  در زمان  $t$  را مشخص می‌کنند. تأخیر ارتباطی به علت ارسال درخواست به سمت کنترل‌کننده و بازگشت نتیجه از آن به سوئیچ ۲ بار محاسبه گردیده است. بدین ترتیب، تأخیر لایه کنترل یعنی میانگین تأخیر برای مجموعه کنترل‌کننده‌های برگزیده شده با (۴) مشخص می‌گردد.

$$CD^t = \frac{1}{p} \sum_{i=1}^p CD_i^t \quad (4)$$

#### ۳-۲-۳ تأخیر میان‌دامنه‌ای

تأخیر میان‌دامنه‌ای در یک دامنه که توسط کنترل‌کننده  $C_i$  پشتیبانی می‌گردد حاصل از میانگین تأخیر انتشار تمامی سوئیچ‌های متعلق به این دامنه در زمان  $t$  است و با نماد  $ID_i^t$  نمایش داده می‌شود.

$$ID_i^t = \sum_{v_e, v_m \in v_i} \frac{A(v_e, v_m) \Psi(v_e, v_m)}{\xi_i} \quad (5)$$

همان‌طور که در شکل ۱ نمایش داده شده است، معماری پیشنهادی شامل ۴ لایه می‌باشد. لایه داده در انتهای این معماری شامل ادوات قابل برنامه‌ریزی جلورانی<sup>۱</sup> برای پردازش بسته‌ها است. این ادوات هیچ هوشمندی در جهت اخذ تصمیم نداشته و مبتنی بر لایه کنترل عمل می‌کنند. این لایه به چند دامنه تقسیم شده که هر دامنه با کنترل‌کننده تخصیص داده شده به خود پشتیبانی می‌گردد.

لایه کنترل به عنوان عنصر هوشمند شبکه شناخته شده و دید کلی از شبکه را در اختیار دارد. هر کنترل‌کننده در این لایه دانشی جامع از کل شبکه را دارا نمی‌باشد و تنها مسئول بخشی از شبکه یعنی دامنه مربوط به خود است. همچنین هر کنترل‌کننده با دیگران کنترل‌کننده‌ها در این لایه در ارتباط است.

لایه مدیریت از درستی عملکرد عملیات‌های صورت گرفته در شبکه اطمینان حاصل می‌کند. این اقدام به واسطه جمع‌آوری آمارهایی از لایه کنترل به دست می‌آید و در راستای تحلیل و بررسی شبکه است. تحلیل‌ها و آمارهای تهیه شده و به لایه بالاتر یعنی لایه دانش ارسال می‌شود. به منظور عدم تأثیر از کارآمدی لایه کنترل، پردازش برقراری توزیع شده کنترل‌کننده‌ها در شبکه نیاز است به صورت خودکار انجام گرفته و توسط یک ماژول از صحت آن اطمینان حاصل شود. این اقدام توسط لایه دانش انجام می‌گیرد. بنابراین عامل یادگیری تقویتی که الگوریتم حل مسئله است در این لایه پیاده‌سازی می‌شود.

### ۳-۲ مدل‌سازی مسئله

شبکه مبتنی بر نرم‌افزار توسط یک گراف بدون جهت مدل می‌شود.  $G = (V, E)$  نمایانگر شبکه است که در آن  $V = \{v_j\}$  مجموعه سوئیچ‌ها بوده و تعداد کل سوئیچ‌ها با  $|V| = k$  مشخص می‌شود. همچنین،  $E$  مجموعه پیوندهای میان سوئیچ‌ها می‌باشد. لایه کنترل شامل مجموعه کنترل‌کننده‌ها است که با  $C = \{c_i\}$  نمایش داده می‌شود. همچنین،  $|C| = n$  تعداد کنترل‌کننده‌ها را مشخص می‌کند. از آنجایی که رویکرد جایابی پیشنهادی چندهدفه است در ادامه به تبیین متغیرهای مربوط به این اهداف می‌پردازیم.

#### ۳-۲-۱ بار لایه کنترل

بار کنترل‌کننده  $C_i$  در زمان  $t$  نمایانگر بار حاصل از تمام سوئیچ‌های متعلق به یک دامنه است که توسط این کنترل‌کننده مورد پشتیبانی قرار می‌گیرند و با نماد  $CL_i^t$  نمایش داده می‌شود.

$$CL_i^t = \sum_{j=1}^k H_{i,j}^t R_{i,j}^t \lambda_i \quad (1)$$

در (۱)،  $H_{i,j}^t$  نمایانگر ارتباط میان کنترل‌کننده  $C_i$  و سوئیچ  $j$  در زمان  $t$  است. چنانچه این ارتباط برقرار باشد این متغیر ۱ بوده و در غیر این صورت ۰ است.  $R_{i,j}^t$  نیز تعداد درخواست‌های سوئیچ  $j$  به کنترل‌کننده  $C_i$  در زمان  $t$  را نمایش می‌دهد. بخشی از منابع مصرف شده برای پردازش یک درخواست در کنترل‌کننده  $C_i$  از نظر پردازنده و حافظه نیز با متغیر  $\lambda_i$  مشخص می‌شود.

پارامترهای تعریف پایداری در شبکه در نظر گرفته می‌شوند که به صورت (۱۰) است.

$$S = \exp\left(\frac{\sum_{j \in V} \ln(1 - \varphi_j) + \sum_{e_{ij} \in E} \ln(1 - \varphi_{e_{ij}})}{|E| + |V|}\right) \quad (10)$$

هدف این مقاله کمینه کردن معیارهای کارآمدی شامل بار لایه کنترل، تأخیر لایه کنترل، تأخیر میان‌دامنه‌ای، توان عملیاتی و پایداری شبکه شامل گره و پیوند است. به همین منظور، تابع هدف به صورت (۱۱) مدل می‌گردد.

$$\begin{aligned} & \text{Min}(\alpha \times \frac{CL'}{CL_{\max}} + \beta \times \frac{CD'}{CD_{\max}} \\ & + \gamma \times \frac{ID'}{ID_{\max}} + \mu \times (1 - \frac{S}{S_{\max}}) + \rho \times (1 - \frac{IT'}{IT_{\max}})) \end{aligned} \quad (11)$$

همچنین، قیود این تابع به صورت زیر است:

$$\forall (w_u, w_v) \in W^2 : w_u \cap w_v = \emptyset \quad (12)$$

$$|W| = p \quad (13)$$

$$\begin{aligned} & \forall (i, l) \in [1, p]^2, \forall j \in [1, k] : \\ & (d_{i,j} < d_{l,j}) \Rightarrow (H_{i,j}^l > H_{l,j}^i) \end{aligned} \quad (14)$$

از آنجایی که تابع هدف شامل معیارهای متفاوت با واحدهای اندازه‌گیری متفاوت است، به منظور نرمال کردن معیارها هر کدام بر مقدار بیشینه خود تقسیم می‌شود. همچنین،  $\alpha, \beta, \gamma, \mu$  و  $\rho$  فاکتورهای وزنی قابل تنظیم هستند که میزان اهمیت هر یک از معیارهای هدف را تعیین می‌کنند و مجموع آن‌ها برابر با ۱ است.

قید (۱۲) به این معنی است که دامنه‌های مشخص شده با یکدیگر همپوشانی نداشته و هر گره به تنها یک دامنه تعلق دارد.

قید (۱۳) تضمین می‌کند تعداد کنترل‌کننده‌های برگزیده شده با تعداد دامنه‌ها مطابقت دارد.

قید (۱۴) نیز تضمین می‌کند که یک سوئیچ باید به نزدیک‌ترین کنترل‌کننده نگاشت داده شود.

به منظور حل مسئله مدل‌سازی شده از یادگیری تقویتی عمیق استفاده می‌کنیم که در بخش بعدی مورد تشریح قرار می‌گیرد.

### ۴- روش حل مسئله

به منظور ارائه روش حل مسئله مدل‌شده پیشنهادی برای جایابی کنترل‌کننده‌ها مبتنی بر چند هدف، در این بخش از رویکردی مبتنی بر یادگیری تقویتی استفاده می‌شود. استفاده از روش شبکه Q عمیق (DQN) برای حل مسئله جایابی کنترل‌کننده‌ها که یک مسئله NP-hard است؛ از نرخ پویایی بالا در شبکه نشأت گرفته است. این رویکرد با جستجو در فضای حالت مسئله می‌تواند به سرعت با پویایی شبکه سازگار شود زیرا دانش خود را از بازخورد به دست آمده از محیط به دست می‌آورد. در ادامه به معرفی عامل DQN برای حل این مسئله و تعیین مکان بهینه کنترل‌کننده‌ها می‌پردازیم.

در (۵)،  $A(v_e, v_m)$  دال بر تأخیر میان ۲ سوئیچ  $v_e$  و  $v_m$  در یک دامنه که توسط کنترل‌کننده  $C_i$  پشتیبانی می‌گردد و این دامنه با  $w_i$  نمایش داده می‌شود، دارد. همچنین،  $\psi(v_e, v_m)$  یک متغیر دودویی نمایانگر وجود یا عدم وجود ارتباط میان دو سوئیچ است. چنانچه این متغیر ۱ باشد دو سوئیچ با یکدیگر داخل دامنه در ارتباط هستند و در غیر این صورت این متغیر مقدار ۰ را به خود می‌گیرد.  $k_i$  نیز، تعداد پیوندها در دامنه  $w_i$  را مشخص می‌کند. بدین ترتیب، تأخیر میان‌دامنه‌ای برای همه دامنه‌های پشتیبانی‌شده توسط مجموعه کنترل‌کننده‌های برگزیده شده به صورت زیر مشخص می‌گردد.

$$ID^t = \frac{1}{p} \sum_{i=1}^p ID_i^t \quad (6)$$

### ۳-۲-۴- توان عملیاتی

توان عملیاتی میان‌دامنه‌ای وابسته به این موضوع است که چه مقدار داده توسط همه دامنه‌های پشتیبانی‌شده توسط مجموعه کنترل‌کننده‌های برگزیده شده در یک برش زمانی مشخص منتقل شده است. چنانچه توان عملیاتی برای یک دامنه مشخص را با  $IT_i^t$  نمایش دهیم، توان عملیاتی برای همه دامنه‌ها توسط (۷) تعیین می‌گردد

$$IT^t = \frac{1}{p} \sum_{i=1}^p IT_i^t \quad (7)$$

### ۳-۲-۵- پایداری شبکه

منظور از پایداری شبکه به حداقل رسیدن احتمال از دست رفتن اطلاعات و داده‌ها در شبکه است. در این راستا، هزینه شکست برای هر سوئیچ و هزینه شکست برای پیوندها مورد محاسبه قرار می‌گیرد تا در انتها معیار پایداری شبکه منتج گردد.

در هر توپولوژی شبکه، درجه مرکزیت یک گره بهترین معیار برای توصیف محوریت آن گره است. هرچه درجه مرکزیت یک گره بیشتر باشد، آن گره تأثیر بیشتری در شبکه داشته و حیاتی‌تر است. هزینه شکست یک گره براساس درجه مرکزیت آن در شبکه می‌تواند مقادیر متفاوتی به خود بگیرد. بر این اساس، هزینه شکست هر گره یا سوئیچ در یک دامنه  $w_i$  به صورت (۸) محاسبه می‌گردد

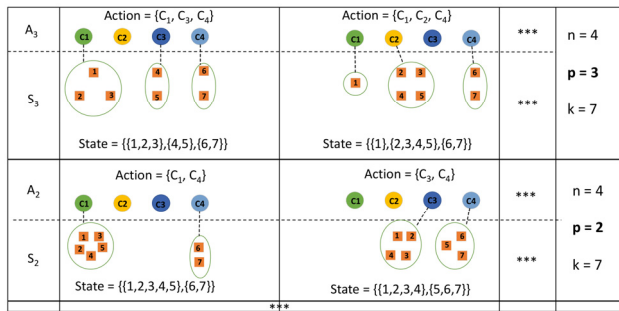
$$\varphi_j = \frac{\sigma_j}{\sum_{v_m, v_e \in w_i} \sigma_m} \quad (8)$$

در (۸)،  $\sigma_j$  درجه مرکزیت سوئیچ  $j$  و  $\sigma_m$  درجه مرکزیت هر سوئیچ دیگری متعلق به همین دامنه را مشخص می‌کند.

به علاوه، درجه مرکزیت یک پیوند اهمیت آن را در شبکه مشخص می‌کند. این درجه مرکزیت به عنوان نسبت تعداد مسیرهایی در دامنه که از این پیوند خاص عبور می‌کنند به تعداد کل کوتاه‌ترین مسیرهایی دامنه تعریف می‌گردد. هرچه درجه مرکزیت یک پیوند بیشتر باشد، نقش آن در شبکه حیاتی‌تر است. بنابراین، هزینه شکست آن پیوند مبتنی بر درجه مرکزیت آن به صورت (۹) محاسبه می‌شود

$$\varphi_{e_{ij}} = \sum_{v_j, v_j' \in w_i} \frac{\sigma_{j,j'}(e)}{\sigma_{j,j'}} \quad (9)$$

در (۹)،  $\sigma_{j,j'}(e)$  کوتاه‌ترین مسیرهایی که از طریق پیوند  $e$  عبور دارند را مشخص کرده و  $\sigma_{j,j'}$  تعداد کل کوتاه‌ترین مسیرهها را نمایش می‌دهد. هزینه شکست سوئیچ و هزینه شکست پیوند به عنوان



شکل ۲: نمایش فضای حالت‌ها و اقدامات برای یک توپولوژی نمونه.

**Input:**  $S = [S_1, \dots, S_n], A = [A_1, \dots, A_n]$

$Max\_reward \leftarrow \cdot$

$p \leftarrow 1$

**while**  $p \leq n$  **do**

**for each**  $state \in S_p$  **then**

$reward, Action \leftarrow DQN(state, S_p, A_p)$

**if**  $reward > Max\_reward$  **then**

$Max\_reward \leftarrow reward$

$Select\_St \leftarrow state$

$Select\_Act \leftarrow Action$

$Select\_p \leftarrow p$

**end if**

**end for**

$p \leftarrow p + 1$

**end while**

**Output:**  $Select\_St, Select\_Act, Select\_p$

شکل ۳: الگوریتم ۱، جایابی کنترل‌کننده‌ها.

به منظور ارائه جزئیات بیشتر، مسئله به صورت گرافیکی در یک توپولوژی کوچک با ۷ سوئیچ در لایه داده و ۴ کنترل‌کننده در لایه کنترل در شکل ۲ نمایش داده شده است. با توجه به شکل، هنگامی که لایه داده به ۳ دامنه تقسیم می‌شود، تنها ۳ کنترل‌کننده برای پشتیبانی از دامنه‌ها مورد استفاده قرار می‌گیرند و مابقی کنترل‌کننده‌ها نمونه‌سازی نمی‌گردند. تعداد کنترل‌کننده‌های برگزیده شده با تعداد دامنه‌ها مطابقت دارد و می‌تواند مقدار ۱ تا  $n$  برابر با تعداد کل کنترل‌کننده‌ها را به خود بگیرد.

از آنجایی که حالت  $S_{t,p}$  و اقدام  $A_{r,p}$  ساختار متفاوتی به ازای هر یک از مقادیر  $p$  دارند؛ داده‌های آموزشی براساس مقادیر  $p$  مورد تقسیم قرار می‌گیرد. سپس، عامل DQN به صورت جداگانه برای هر مقدار  $p$  مورد آموزش قرار می‌گیرد ( $DQN(S_{t,p}, A_{r,p})$ ). برای تعیین بهترین نگاشت میان مجموعه حالت‌ها (دامنه‌ها، خوشه‌های سوئیچ‌ها) و مجموعه کنترل‌کننده‌ها مطابق الگوریتم ۱ عمل می‌کنیم (شکل ۳). این الگوریتم مجموعه حالت‌های تقسیم‌شده و مجموعه اقدامات تقسیم‌شده بر اساس  $p$  را به عنوان ورودی می‌گیرد. سپس، با استفاده از عامل DQN آموزش داده شده، اقدام متناظر با حداکثر پاداش را به صورت تکرار شونده برای هر یک از حالت‌های فرعی تعیین می‌نماید. خروجی الگوریتم، تعداد بهینه و شناسه کنترل‌کننده‌هایی است که بایستی استقرار یابند ( $Select\_p, Select\_Act$ ) و مجموعه مربوط از خوشه‌های سوئیچ‌ها ( $Select\_St$ ) می‌باشد.

## ۴-۱ حالت‌ها

در مدل حاضر، حالت بیانگر تقسیم‌بندی فعلی لایه داده میان دامنه‌های کنترلی است. بنابراین، حالت در این مدل ساختاری توپولوژیک دارد و نحوه توزیع سوئیچ‌ها بین کنترل‌کننده‌ها را توصیف می‌کند، نه مقادیر عددی معیارهای عملکردی نظیر تأخیر یا بار.

مجموعه حالت  $S_{t,p}$  مرتبط با تقسیم لایه داده به تعدادی خوشه یا دامنه از سوئیچ‌ها است. در این مجموعه  $p$  دامنه را مشخص می‌کند و  $t$  می‌تواند از مقدار ۱ تا  $T_p$  را به خود بگیرد که مجموعه کل حالت‌های تشکیل‌دهنده خوشه یا دامنه  $p$  می‌باشد. مجموعه حالت  $S_{t,p}$  به صورت (۱۵) تعریف می‌گردد.

$$S_{t,p} = [w_{t,1}, w_{t,2}, \dots, w_{t,p}] \quad (15)$$

بردار  $w_{t,i}$  مشخص‌کننده خوشه‌ای از سوئیچ‌ها است و به صورت  $w_{t,i} = [v_{j,1}, \dots, v_{j,p}]$  تعریف می‌شود. این تعریف به صورتی است که  $\sum_{i=1}^p |w_{t,i}| = k$  بوده و  $k$  تعداد کل سوئیچ‌ها می‌باشد. همچنین، مجموعه کل حالت‌ها مربوط به خوشه یا دامنه  $p$  با  $S_p$  تعریف می‌گردد.

$$S_p = [S_{1,p}, S_{2,p}, \dots, S_{T_p,p}] \quad (16)$$

## ۴-۲ اقدامات

در هر گام زمانی، عامل یک اقدام را انتخاب می‌کند که بیانگر تغییر در تقسیم‌بندی فعلی شبکه است. اعمال اقدام توسط عامل منجر به تولید تقسیم‌بندی جدید می‌شود که در گام بعدی مورد ارزیابی قرار می‌گیرد. بنابراین ویژگی‌هایی مانند تأخیر، بار و پایداری بخشی از تابع هدف و محاسبه‌ی پاداش هستند. در واقع، عامل بر اساس تغییر در این معیارها پس از هر اقدام پاداش دریافت می‌کند تا از طریق یادگیری تقویتی، تقسیم‌بندی بهینه‌ای از شبکه حاصل گردد.

اقدام انجام گرفته توسط عامل، با یک بردار مشخص می‌شود که مجموعه‌ای از کنترل‌کننده‌های برگزیده شده از میان کنترل‌کننده‌های موجود را نمایش داده و به صورت (۱۷) تعریف می‌گردد.

$$A_{r,p} = [c_{r,1}, \dots, c_{r,p}] \quad (17)$$

در (۱۷)،  $r$  نمایانگر تعداد اقدامات است و به این ترتیب مجموعه کل اقدامات مرتبط با دامنه  $p$  به صورت زیر است.

$$A_p = [A_{1,p}, A_{2,p}, \dots, A_{R_p,p}] \quad (18)$$

در (۱۸) نیز  $R_p$  تعداد اقدامات مربوط به دامنه  $p$  را نمایش می‌دهد.

## ۴-۳ پاداش

تابع پاداش  $R$  حاوی کمینه ساختن تابع هدف نرمال شده مسئله چندهدفه جایابی کنترل‌کننده بوده و به صورت زیر تشریح می‌گردد.

$$R = \alpha \times \frac{CL^t}{CL_{max}} + \beta \times \frac{CD^t}{CD_{max}} + \gamma \times \frac{ID^t}{ID_{max}} + \mu \times \left(1 - \frac{S}{S_{max}}\right) + \rho \times \left(1 - \frac{IT^t}{IT_{max}}\right) \quad (19)$$

شایان ذکر است که عامل DQN حاوی تعیین بهترین نگاشت میان مجموعه حالت‌ها و مجموعه اقدامات است در صورتی که  $1/R$  بیشینه گردد (تابع  $R$  کمینه گردد).

جدول ۱: پارامترهای ارزیابی.

مقدار	پارامتر ارزیابی
۳۲	تعداد سوئیچ‌ها
۱۰	تعداد کنترل‌کننده‌ها
۰٫۲	نرخ اکتشاف
۰٫۰۱	نرخ یادگیری
۰٫۶	نرخ تخفیف
۱۰۰۰	تعداد اپیزود
۲۰۰	تعداد گام هر اپیزود
۲۰۰۰	اندازه حافظه
۱	حداقل تعداد سوئیچ در هر دامنه
۱۳	حداکثر تعداد سوئیچ در هر دامنه

شده است، به‌گونه‌ای که مقدار  $\epsilon$  از ۰٫۲ در آغاز آموزش شروع شده است و به صورت خطی کاهش می‌یابد. فرآیند آموزش در مجموع طی ۱۰۰۰ اپیزود انجام شده است که هر دوره شامل ۲۰۰ گام زمانی می‌باشد. معیار توقف آموزش زمانی است که تغییرات میانگین مقادیر  $Q$  در طول ۲۰ دوره‌ی متوالی کمتر از مقدار مشخصی گردد.

### ۵- ارزیابی روش پیشنهادی

به منظور ارزیابی روش پیشنهادی، الگوریتم حل آن به زبان برنامه‌نویسی پایتون پیاده‌سازی شده و اجرا گردیده است. لایه کنترل معماری پیشنهادی توسط مجموعه‌ای از کنترل‌کننده‌ها از نوع ONOS [۲۱] ایجاد شده است. لایه داده نیز متشکل از ۳۲ گره از نوع OpenvSwitch از طریق یک اسکرپت به زبان پایتون [۲۲] در Mininet ایجاد گردیده است. همچنین، ترافیک میان گره‌های شبکه با Iperf منتقل شده است. حجم ترافیک بین گره‌ها بر اساس توزیع یکنواخت تصادفی در بازه‌ی [۱۰، ۱۰۰] Mbps تولید گردید تا شرایط واقعی شبکه‌های پویا شبیه‌سازی شود. از کتابخانه Tensorflow نیز برای استقرار دو شبکه عصبی دارای معماری یکسان برای پیاده‌سازی یادگیری تقویتی عمیق استفاده شده است.

تعداد کل سوئیچ‌ها در لایه داده ۳۲ سوئیچ بوده و تعداد کل کنترل‌کننده‌ها در لایه کنترل ۱۰ می‌باشد. در طی فاز آموزش عامل DQN از روش  $\epsilon$ -Greedy برای انتخاب اقدامات و تعیین سیاست استفاده کرده است. نرخ اکتشاف در این روش ۰٫۲ بوده و پارامترهای شبکه  $Q$  هدف هر ۲۰۰ گام پس از آموزش شبکه  $Q$  اصلی مورد بروزرسانی قرار می‌گیرند. همچنین، نرخ یادگیری ۰٫۰۱ بوده و نرخ تخفیف به مقدار ۰٫۶ تنظیم گردیده است. به‌علاوه، هر فاز از آموزش دارای ۱۰۰۰ اپیزود می‌باشد. برخی از پارامترهای مهم ارزیابی مورد استفاده در شبیه‌سازی‌ها در جدول ۱ نشان داده شده است.

آزمایش‌ها در چند توپولوژی مختلف شبکه نیز انجام شد تا پایداری و تعمیم‌پذیری مدل پیشنهادی ارزیابی شود. از توپولوژی‌های کلاسیک NSFNET، GEANT و GARR استفاده گردید و داده‌های مربوط به ساختار آنها از پایگاه داده‌ی Topology Zoo استخراج شد. برای افزایش پایداری آموزش، الگوریتم در هر توپولوژی به مدت ۱۰۰۰ اپیزود اجرا شد و میانگین نتایج روی ۵ تکرار مستقل گزارش گردید. تمامی نمودارهای ارائه‌شده بر اساس میانگین پنج اجرای مستقل رسم شده‌اند و میله‌های

- 1: **Initialize:** the replay buffer
- 2: **Initialize:** the main deep neural network with random weight  $\theta$
- 3: **Initialize:** the target deep neural network with weight  $\theta' = \theta$
- 4: **for each episode do**
- 5: Obtain the initial observation  $S_1$
- 6: **for each steps of episode do**
- 7: Choose  $a$  with a random probability  $\phi$  as
- 8: **if  $\phi < \epsilon$  then**
- 9: randomly select an action  $a$
- 10: **else**
- 11: select  $a = \arg \max(s, a, \theta)$
- 12: **end if**
- 13: Execute action  $a$ , receive the reward  $r$
- 14: Store the  $(s, a, r)$  into replay buffer
- 15: Calculate the target  $Q$  value
- 16: Update the weight of the main deep neural network by minimizing the  $Loss(\theta)$
- 17: Update the target deep neural network parameter
- 18: **end for**
- 19: **end for**

شکل ۴: الگوریتم ۲، Deep Q-Network.

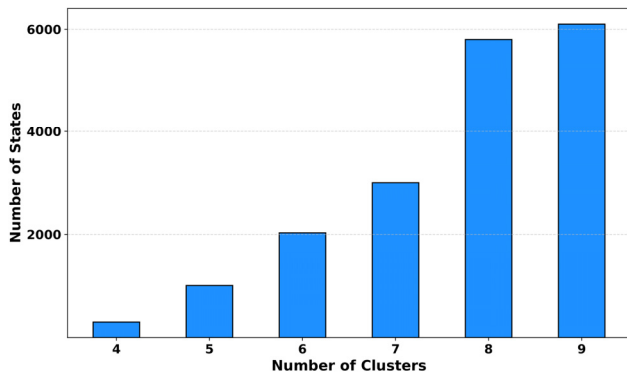
الگوریتم DQN نیز که از شبکه عصبی عمیق در آن بهره‌برداری شده و به دنبال بهینه کردن پارامترها و وزن‌های این شبکه است به صورت شکل ۴ توسعه می‌یابد.

در روش پیشنهادی حاضر، عامل یادگیری تقویتی عمیق با استفاده از یک شبکه‌ی پیش‌خور<sup>۱</sup> به‌منظور محاسبه تابع ارزش اقدام  $Q(s, a, \theta)$  پیاده‌سازی شده است. در این مدل، دو شبکه‌ی هم‌ساختار شامل شبکه‌ی اصلی و شبکه‌ی هدف به‌کار رفته‌اند تا با بهره‌گیری از سازوکار بازپخش تجربه<sup>۲</sup>، پایداری فرآیند یادگیری افزایش یابد.

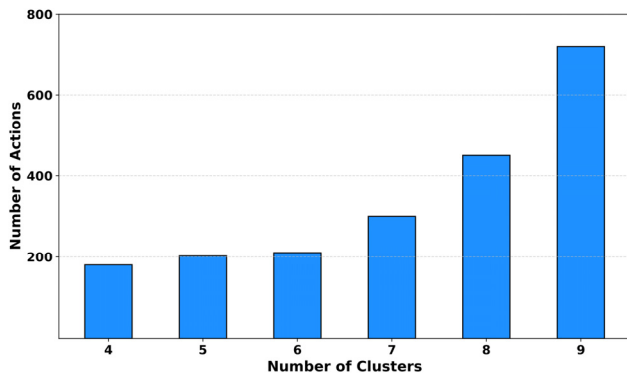
هر شبکه از سه لایه‌ی کاملاً متصل تشکیل شده است. یک لایه‌ی ورودی با ابعادی متناسب با اندازه‌ی فضای حالت تعریف‌شده در مسئله، دو لایه‌ی پنهان با به‌ترتیب ۱۲۸ و ۶۴ نورون، و یک لایه‌ی خروجی با ابعادی برابر با تعداد اقدامات ممکن تعریف‌شده در مسئله است. برای لایه‌های پنهان از تابع فعال‌سازی ReLU و برای لایه‌ی خروجی از تابع فعال‌سازی خطی استفاده شده است.

به‌منظور بهینه‌سازی وزن‌های شبکه از الگوریتم بهینه‌سازی Adam استفاده شده و تابع زیان به‌صورت میانگین مربعات خطا<sup>۳</sup> میان خروجی شبکه و مقدار هدف محاسبه‌شده از رابطه‌ی به‌روزرسانی تعریف گردیده است. نرخ یادگیری برابر با ۰٫۰۱، اندازه‌ی دسته‌ی آموزشی برابر با ۶۴ و اندازه‌ی حافظه‌ی بازپخش برابر با ۲۰۰۰ تجربه در نظر گرفته شده است. پارامترهای شبکه‌ی هدف هر ۲۰۰ گام آموزشی به‌روزرسانی می‌شوند تا از نوسانات و واگرایی در فرآیند یادگیری جلوگیری شود. نرخ تخفیف برابر با ۰٫۶ انتخاب شده تا توازن مناسبی میان پاداش‌های کوتاه‌مدت و بلندمدت برقرار گردد. راهبرد انتخاب اقدام بر اساس سیاست  $\epsilon$ -Greedy تنظیم

1. Feed-Forward
2. Experience Replay
3. Mean Square Error (MSE)



شکل ۵: بررسی تاثیر افزایش تعداد دامنه‌ها بر تعداد حالت‌ها.



شکل ۶: بررسی تاثیر افزایش تعداد دامنه‌ها بر تعداد اقدامات.

میان آن‌ها افزایش می‌یابد.

همچنین، شکل ۶ تاثیر افزایش تعداد دامنه‌ها بر تعداد اقدامات را نمایش می‌دهد. به تبع آنچه برای تعداد حالت‌ها گفته شد، افزایش تعداد دامنه‌ها، افزایش تعداد اقدامات را نیز سبب می‌گردد زیرا، هر دامنه جدید، مجموعه‌ای از انتخاب‌های جدید را به فضای مسئله اضافه می‌کند. در ادامه ارزیابی عملکرد معیارها تنها برای تعداد دامنه با مقدار ۴، ۵ و ۶ مورد بررسی قرار می‌گیرند.

### ۳-۵ بررسی تاثیر تعداد دامنه‌ها بر معیارهای هدف

شکل ۷ بررسی تاثیر تعداد دامنه‌های انتخابی و در واقع تعداد کنترل‌کننده‌ها بر معیار تأخیر لایه کنترل را نمایش می‌دهد. با توجه به شکل، با افزایش تعداد کنترل‌کننده‌ها در هر دو روش پیشنهادی و کار مرتبط، تأخیر لایه کنترل با کاهش مواجه گشته است. این مسئله به این دلیل است که با انتخاب تعداد بیشتری از کنترل‌کننده‌ها برای پشتیبانی از دامنه‌های ایجاد شده در لایه داده، سوئیچ‌های موجود در این لایه به کنترل‌کننده نزدیک‌تری به خود از میان کنترل‌کننده‌های برگزیده، متصل گردیده‌اند. بر همین اساس، تأخیر ارتباط میان سوئیچ و کنترل‌کننده کاهش یافته و به تبع آن تأخیر لایه کنترل نیز کاهش می‌یابد. همچنین، به علت افزایش تعداد کنترل‌کننده‌های برگزیده، بار سوئیچ‌ها میان تعداد کنترل‌کننده‌های بیشتری پخش شده و با توازن بار ایجاد شده تأخیر پردازش در کنترل‌کننده نیز با کاهش مواجه خواهد شد. مطابق شکل، رویکرد پیشنهادی به نسبت کار مرتبط عملکرد بهتری از نظر معیار تأخیر لایه کنترل دارد. این مسئله از این جهت است که با افزودن معیار پایداری به اهداف مسئله، کنترل‌کننده‌ها با هزینه شکست کمتر و پایداری بالاتر برای پشتیبانی از دامنه‌ها برگزیده شده و این امر سبب کاهش تأخیر لایه

خطا نشان‌دهنده‌ی بازه‌ی اطمینان<sup>۱</sup> نتایج می‌باشند.

روش پیشنهادی با کار [۸] مورد مقایسه قرار می‌گیرد. این کار یکی از محدود روش‌هایی است که از یادگیری تقویتی عمیق برای حل مسئله جایابی کنترل‌کننده‌ها بهره یافته است و به همین دلیل به عنوان کار مورد مقایسه برگزیده شده است. تفاوت دو مسئله در اهداف مورد بررسی می‌باشد. معیارهای ارزیابی عملکرد روش پیشنهادی در مقایسه با روش مورد مقایسه شامل بررسی تاثیر تعداد دامنه‌ها بر معیارهای هدف یعنی بار لایه کنترل، تأخیر لایه کنترل، تأخیر میان‌دامنه‌ای، توان عملیاتی میان‌دامنه‌ای و پایداری شبکه است.

همچنین، از یک روش مبتنی بر تخصیص حریصانه نیز برای مقایسه استفاده گردید؛ در این روش هر سوئیچ به نزدیک‌ترین کنترل‌کننده بر اساس کمینه‌سازی تأخیر اختصاص می‌یابد.

### ۱-۵ بررسی پیچیدگی محاسباتی الگوریتم پیشنهادی

از آنجا که مسئله جایابی کنترل‌کننده در شبکه‌های نرم‌افزارمحور یک مسئله NP-hard است، روش‌های دقیق برای شبکه‌های مقیاس بزرگ از کارایی کافی برخوردار نیستند. در روش پیشنهادی مبتنی بر DQN، از یادگیری تقویتی برای تقریب تابع تصمیم بهینه استفاده شده است. پیچیدگی محاسباتی الگوریتم را می‌توان به دو بخش اصلی تقسیم کرد. بخش اول به تولید داده‌های موردنیاز برای بازپخش تجربه مربوط است. بخش دوم نیز با به‌روزرسانی شبکه عصبی ارتباط دارد.

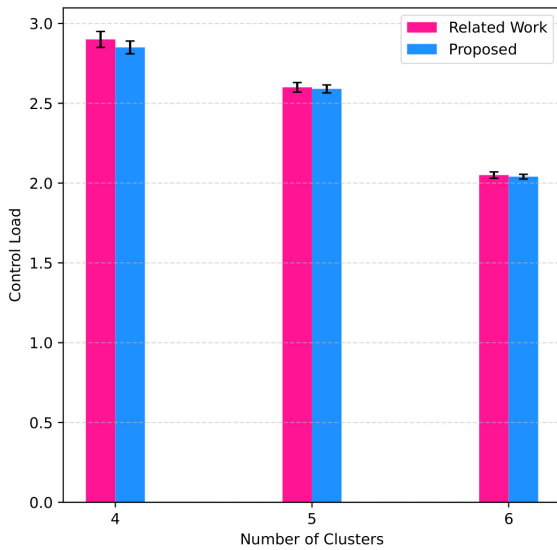
در هر گام آموزش، عامل DQN یک اقدام از میان  $|A|$  حالت ممکن انتخاب می‌کند و با انجام آن، وضعیت جدید و پاداش متناظر محاسبه می‌شود. بنابراین پیچیدگی مربوط به تولید تجربه در هر گام برابر با  $O(|A|)$  است. در بخش یادگیری، هر مرحله‌ی به‌روزرسانی شبکه شامل محاسبه‌ی گرادیان و اصلاح وزن‌ها است که با توجه به دو لایه‌ی پنهان با به ترتیب ۱۲۸ و ۶۴ نورون، پیچیدگی آن به صورت تقریبی  $O(128 \times 64)$  خواهد بود. با در نظر گرفتن E به عنوان تعداد کل اپیزودها و T به عنوان تعداد گام‌ها در هر اپیزود، پیچیدگی کلی الگوریتم  $O(E \times T \times (|A| + 128 \times 64))$  خواهد بود.

این مرتبه زمانی در عمل از مرتبه‌ی چندجمله‌ای بوده و با توجه به ماهیت تقریبی یادگیری تقویتی، بسیار کمتر از روش‌های دقیق مانند ILP است. از نظر حافظه، نگهداری بافر بازپخش با اندازه M نمونه، نیازمند فضای  $O(M)$  است که قابل مدیریت بوده است.

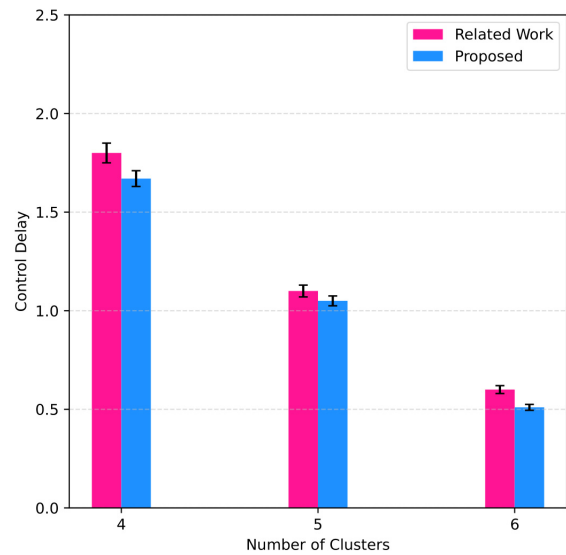
بدین ترتیب، روش پیشنهادی با حفظ دقت تصمیم‌گیری، از پیچیدگی محاسباتی قابل‌قبولی برخوردار است و در شبکه‌های با اندازه متوسط و بزرگ نیز قابل استفاده است.

### ۲-۵ بررسی تاثیر تعداد دامنه‌ها بر تعداد حالت‌ها و اقدامات

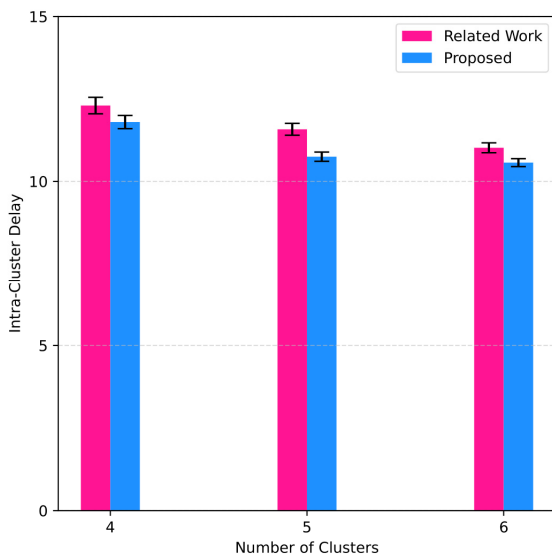
با تحلیل تاثیر انتخاب تعداد دامنه‌ها بر تعداد حالت‌ها و اقدامات در یادگیری Q، به انتخاب تعداد p مناسب برای بررسی سایر معیارهای ارزیابی در زیربخش بعدی دست می‌یابیم. شکل ۵ نمایانگر تاثیر افزایش تعداد دامنه‌ها یا خوشه‌ها بر تعداد حالت‌ها است. با افزایش دامنه‌ها حالت‌های سیستم نیز افزایش می‌یابد. این افزایش از آنجایی است که هر دامنه دارای مجموعه‌ای از سوئیچ‌ها است و با افزایش تعداد دامنه‌ها، تعداد حالت‌های ممکن برای ترکیب چینش در لایه داده و توزیع کنترل‌کننده‌ها



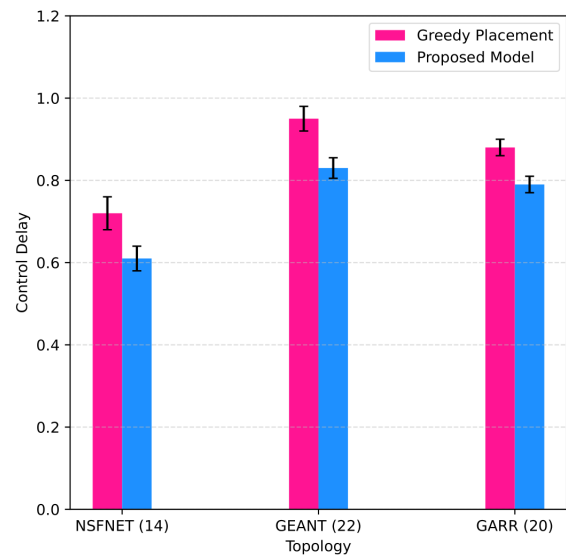
شکل ۹: بررسی تاثیر تعداد دامنه‌ها بر بار لایه کنترل.



شکل ۷: بررسی تاثیر تعداد دامنه‌ها بر تأخیر لایه کنترل.



شکل ۱۰: بررسی تاثیر تعداد دامنه‌ها بر تأخیر درون‌دامنه‌ای.



شکل ۸: بررسی تاثیر انواع توپولوژی بر میانگین تأخیر لایه کنترل.

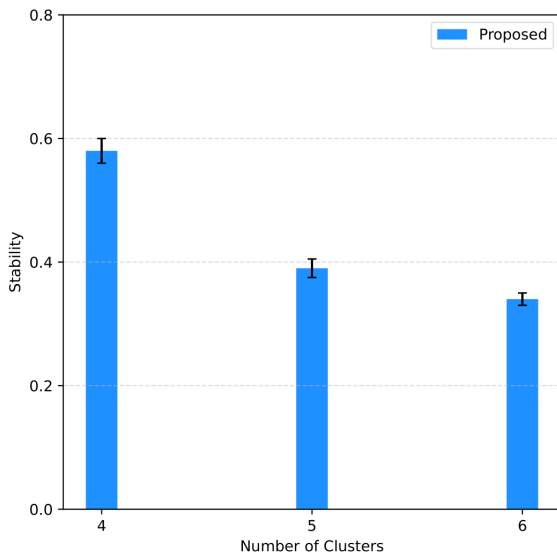
شکل ۹ بررسی تاثیر تعداد دامنه انتخابی بر معیار بار لایه کنترل را نمایش می‌دهد. با افزایش تعداد کنترل‌کننده‌ها در هر دو روش پیشنهادی و مورد مقایسه، بار لایه کنترل کاهش یافته است زیرا بار لایه داده با افزایش تعداد کنترل‌کننده‌ها توزیع بیشتری دارد. همچنین، روش پیشنهادی نسبت به روش مورد مقایسه در معیار بار لایه کنترل عملکرد تقریباً یکسانی دارد.

شکل ۱۰ نیز بررسی تاثیر تعداد دامنه انتخابی بر تأخیر درون‌دامنه‌ای را نمایش می‌دهد. با افزایش تعداد کنترل‌کننده‌ها در هر دو روش پیشنهادی و مورد مقایسه، این تأخیر با کاهش مواجه بوده است. زیرا با انتخاب تعداد بیشتری کنترل‌کننده، با وجود تعداد ثابتی سوئیچ در لایه داده، دامنه‌ها از نظر تعداد سوئیچ کوچکتر شده و در نتیجه میانگین تأخیر انتشار میان سوئیچ‌های موجود در یک دامنه کاهش می‌یابد. همچنین، رویکرد پیشنهادی به نسبت کار مرتبط در معیار تأخیر میان‌دامنه‌ای عملکرد بهتری دارد. زیرا با افزودن معیار پایداری به اهداف، پیوندها با هزینه شکست کمتر و پایداری بالاتر برای وجود در کوتاه‌ترین مسیرها میان یک دامنه انتخاب شده و این امر سبب کاهش تأخیر میان‌دامنه‌ای گشته است.

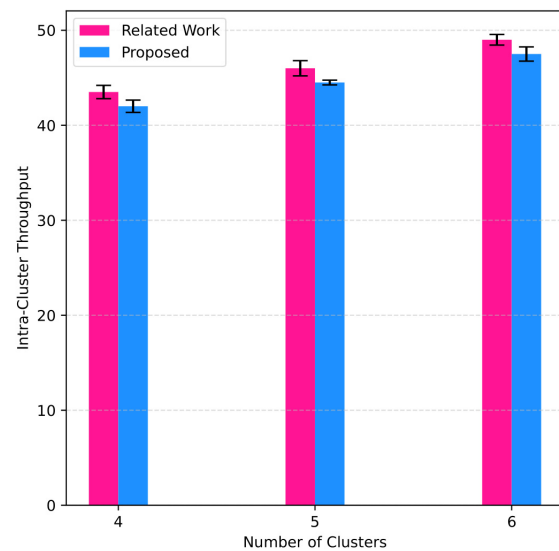
شکل ۱۱ بررسی تاثیر تعداد دامنه‌های انتخابی بر معیار توان عملیاتی درون‌دامنه‌ای را نمایش می‌دهد. با افزایش تعداد کنترل‌کننده‌ها در هر دو

کنترل گشته است.

شکل ۸ تأثیر نوع توپولوژی شبکه بر میانگین تأخیر لایه کنترل را در دو روش پیشنهادی و روش حریصانه نشان می‌دهد. همان‌طور که در شکل مشاهده می‌شود، با افزایش اندازه و پیچیدگی توپولوژی شبکه مقدار تأخیر لایه کنترل در هر دو روش روند افزایشی دارد. این افزایش به دلیل افزایش تراکم ترافیک در شبکه‌های بزرگ‌تر است. با این حال، در تمامی توپولوژی‌ها، روش پیشنهادی نسبت به روش حریصانه، تأخیر کمتری را نشان می‌دهد. این برتری ناشی از در نظر گرفتن هم‌زمان معیارهای چندگانه در فرآیند جایابی کنترل‌کننده‌ها، از جمله توازن بار، پایداری شبکه و کاهش تأخیر میان‌دامنه‌ای است؛ عواملی که باعث می‌شوند سوئیچ‌ها به کنترل‌کننده‌هایی با ظرفیت بالاتر و موقعیت مکانی بهینه‌تر متصل شوند. در توپولوژی GEANT که دارای بیشترین تعداد گره و لینک است، اختلاف عملکرد میان دو روش محسوس‌تر بوده و روش پیشنهادی حدود ۱۳٪ کاهش در تأخیر لایه کنترل نسبت به روش حریصانه ارائه می‌دهد. به‌طور کلی، نتایج این شکل بیانگر آن است که رویکرد پیشنهادی از کارایی بالاتری برخوردار بوده و توانایی تعمیم به توپولوژی‌های متنوع را داراست.



شکل ۱۲: بررسی تاثیر تعداد دامنه‌ها بر پایداری شبکه.



شکل ۱۱: بررسی تاثیر تعداد دامنه‌ها بر توان عملیاتی درون‌دامنه‌ای.

اگرچه نتایج به‌دست‌آمده نشان‌دهنده کارایی بالای روش پیشنهادی در بهینه‌سازی جایابی کنترل‌کننده‌ها و کاهش تأخیر شبکه است، اما مانند سایر رویکردهای مبتنی بر یادگیری ماشین، این روش دارای برخی محدودیت‌ها نیز می‌باشد. از جمله می‌توان به افزایش فضای حالت و فضای اقدام در شبکه‌های بسیار بزرگ اشاره کرد که ممکن است زمان همگرایی عامل یادگیرنده را افزایش دهد. همچنین مدل DQN در برابر نویز یا داده‌های غیرایستا حساس است و در شرایط واقعی نیاز به بازآموزی یا تنظیم پویا دارد.

در راستای توسعه‌ی این پژوهش، در نظر داریم مدل پیشنهادی را با افزودن معیارهای دیگری همچون اطمینان‌پذیری و امنیت پیاده‌سازی و ارزیابی نماییم. بررسی تأثیر جایابی کنترل‌کننده‌ها بر معیارهای امنیتی شبکه، از جمله مقاومت در برابر حملات انکار سرویس توزیع‌شده (DDoS)، می‌تواند درک عمیق‌تری از ارتباط بین ساختار لایه کنترل و تاب‌آوری امنیتی SDN فراهم کند. علاوه بر این، بهره‌گیری از یادگیری تقویتی چندعامله می‌تواند مقیاس‌پذیری و توان تصمیم‌گیری توزیع‌شده را بهبود بخشد و موجب افزایش کارایی در شبکه‌های بزرگ شود. به‌کارگیری معماری سلسله‌مراتبی برای لایه‌ی کنترل با در نظر گرفتن کنترل‌کننده‌ی پشتیبان نیز از دیگر مسیرهای توسعه است که می‌تواند پایداری و تاب‌آوری لایه‌ی کنترل را افزایش دهد. همچنین، پیاده‌سازی و ارزیابی مدل پیشنهادی در محیط واقعی با استفاده از کنترل‌کننده‌ی OpenDayLight (ODL) به‌عنوان یکی از نمونه‌های صنعتی و متن‌باز کنترل‌کننده‌ها، از دیگر برنامه‌های آتی این پژوهش خواهد بود.

## مراجع

- [1] C. Urrea and D. Benítez, "Software-defined networking solutions, architecture and controllers for the industrial internet of things: A review," *Sensors*, vol. 21, no. 19, Article ID: 6585, 2021.
- [2] Y. Maleh, Y. Qasmaoui, K. El Gholami, Y. Sadqi, and S. Mounir, "A comprehensive survey on sdn security: threats, mitigations, and future directions," *Journal of Reliable Intelligent Environments*, vol. 9, no. 2, pp.201-239, 2023.
- [3] A. N. Alhaj and N. Dutta, "Analysis of security attacks in sdn network: A comprehensive survey," *Contemporary Issues in Communication, Cloud and Big Data Analytics*, pp.27-37, 2022.
- [4] K. Nisar, et al., "A survey on the architecture, application, and security of software defined networking: Challenges and open issues," *Internet of Things*, vol. 12, Article ID: 100289, 2020.
- [5] S. Ahmad and A. H. Mir, "Scalability, consistency, reliability and security in sdn controllers: a survey of diverse sdn controllers,"

روش، توان عملیاتی افزایش یافته است. زیرا اندازه دامنه‌ها از نظر تعداد سوئیچ کاهش یافته و با کاهش پیوندها میزان داده منتقل‌شده با افزایش روبرو شده است. همچنین، رویکرد پیشنهادی در تعداد دامنه بالاتر عملکرد بهتری به نسبت کار مورد مقایسه دارد. زیرا با افزودن معیار پایداری به اهداف مسئله، پیوندها با درجه مرکزیت کمتر و هزینه شکست کمتر برگزیده شده و این امر سبب افزایش داده‌های منتقل‌شده در دامنه می‌گردد.

شکل ۱۲ نیز بررسی تاثیر تعداد دامنه انتخابی بر معیار پایداری را نمایش می‌دهد. با افزایش تعداد دامنه در روش پیشنهادی، هزینه شکست گره و پیوند کاهش یافته و این به معنی افزایش پایداری در شبکه است. این مسئله به این دلیل است که با افزایش تعداد دامنه‌ها، تعداد سوئیچ‌های مرتبط به هر دامنه کاهش یافته و از میان پیوندهای موجود آنانی که هزینه شکست کمتری دارند برگزیده می‌شوند. شایان ذکر است که مرجع [۸n] فاقد مدل‌سازی صریح برای پایداری بوده و در آن هیچ مکانیزمی برای انتخاب گره‌ها و پیوندها با درجه مرکزیت پایین‌تر لحاظ نشده است.

## ۶- نتیجه‌گیری

در این مقاله، مسئله چند هدفه جایابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار مدل‌سازی شده است. در مدل پیشنهادی، اهداف مورد بررسی شامل تأخیر لایه کنترل، بار لایه کنترل، تأخیر میان‌دامنه‌ای، توان عملیاتی و پایداری است. همچنین، به منظور حل مدل پیشنهادی در مدت زمان قابل قبول، یک الگوریتم مبتنی بر یادگیری تقویتی عمیق استفاده شده است. در این الگوریتم، فضای حالت‌ها، فضای اقدامات و همچنین تابع پاداش متناسب با مسئله تعریف گشته و عامل یادگیری در جهت بیشینه کردن طولانی مدت پاداش (کمینه کردن هدف) عمل می‌نماید. برخلاف کارهای پیشین همچون مرجع [۸] که تنها به برخی اهداف مانند تأخیر و تعادل بار پرداخته‌اند، در این مقاله مجموعه‌ای گسترده از اهداف از جمله پایداری گره و پیوند، به‌طور ترکیبی و دقیق مدل شده است که این مسئله از جنبه مدل‌سازی و نتایج به دست آمده تفاوت قابل توجهی با پژوهش‌های قبلی ایجاد کرده است. نتایج شبیه‌سازی نشان داده‌اند که روش پیشنهادی معیارهای هدف از جمله تأخیر و توان عملیاتی را در مقایسه با کار مرتبط به‌طور قابل توجهی بهبود می‌بخشد.

- Physical Systems*, Dubai, United Arab Emirates, pp. 331-336, 10-12 Jun. 2025.
- [18] B. Li, *et al.*, "DDPG-based load-aware QoS guaranteed SDN controller placement for Internet of vehicles," *IEEE Internet of Things Journal*, vol. 12, no. 24, 15 Dec. 2025.
- [19] A. Kumari, A. Roy and A. Singh Sairam, "Optimizing SDN controller load balancing using online reinforcement learning," *IEEE Access*, vol. 12, pp. 131591-131604, 2024.
- [20] M. Farhan, *et al.*, "ReCAP: Reliability-capacity aware joint controller placement and routing using a hybrid AI approach," *IEEE Trans. on Reliability*, vol. 74, no. 4, pp. 5686- 5700, 2025.
- [21] J. Hyun and J. W. Hong, "Knowledge-defined networking using in-band network telemetry," in *Proc. 19th Asia-Pacific Network Operations and Management Symp.*, pp. 54-57, Seoul, South Korea, 27-29 Sept. 2017.
- [22] P.T.A. Quang, Y. Hadjadj-Aoul, and A. Outtagarts, "A deep reinforcement learning approach for VNF forwarding graph embedding," *IEEE Trans. on Network and Service Management*, vol. 16, no 4, pp. 1318-1331, Dec. 2019.
- مبین قلی‌زاده** مدرک کارشناسی مهندسی برق را از دانشگاه گیلان و مدرک کارشناسی ارشد مهندسی کامپیوتر گرایش شبکه‌های کامپیوتری را از دانشگاه علم و صنعت ایران به ترتیب در سال‌های ۱۳۹۹ و ۱۴۰۳ دریافت نمود. عنوان پایان‌نامه کارشناسی ارشد وی «یک رویکرد یادگیری ماشین چندهدفه برای مکان‌یابی کنترل‌کننده‌ها در شبکه‌های مبتنی بر نرم‌افزار» بوده است. وی از سال ۱۴۰۴ دانشجوی دکتری مهندسی کامپیوتر گرایش شبکه و رایانش در دانشکده مهندسی کامپیوتر دانشگاه علم و صنعت ایران می‌باشد. زمینه‌های پژوهشی مورد علاقه ایشان شامل شبکه‌های مبتنی بر نرم‌افزار، یادگیری ماشین در شبکه‌های کامپیوتری، تخلیه‌بار محاسباتی و معماری‌های نوین شبکه است.
- ناصر مزینی** مدرک کارشناسی مهندسی برق را در سال ۱۳۶۹ از دانشگاه صنعتی شریف دریافت نمود. وی در سال ۱۳۷۳ موفق به اخذ مدرک کارشناسی ارشد انفورماتیک (سیستم‌های اطلاعاتی و تلماتیک) از دانشگاه سوپلک فرانسه گردید و در ادامه، مدرک دکتری انفورماتیک را در سال ۱۳۷۷ از دانشگاه رن یک فرانسه اخذ نمود. دکتر مزینی هم‌اکنون دانشیار رشته مهندسی کامپیوتر با تخصص هوش مصنوعی می‌باشد. زمینه‌های پژوهشی مورد علاقه ایشان شامل محاسبات نرم، سیستم‌های چندعامله، شبکه‌های کامپیوتری، شبکه‌های مبتنی بر نرم‌افزار، شبکه‌های هوشمند برق و آموزش الکترونیکی است.
- Journal of Network and Systems Management*, vol. 29, pp.1-59, 2021.
- [6] B. Isong, R. R. S. Molose, A. M. Abu-Mahfouz, and N. Dladlu, "Comprehensive review of sdn controller placement strategies," *IEEE Access*, vol. 8, pp.170070-170092, 2020.
- [7] M. Rahouti *et al.*, "SDN security review: Threat taxonomy, implications, and open challenges," *IEEE Access*, vol. 10, pp. 45820-45854, 2022.
- [8] E. H. Bouzidi, A. Outtagarts, R. Langar, and R. Boutaba, "Dynamic clustering of software defined network switches and controller placement using deep reinforcement learning," *Computer Networks*, vol.207, Article ID: 108852, 2022.
- [9] K. Saeed and M. O. Ullah, "Toward reliable controller placements in software-defined network using constrained multi-objective optimization technique," *IEEE Access*, vol. 10, pp.129865-129883, 2022.
- [10] C. Liao, *et al.*, "Modexp: A multi objective based approach for solving distributed controller placement problem in software defined network," *Sensors*, vol.22, no.15, Article ID: 5475, 2022.
- [11] M. M. Kazemian and M. Mirabi, "Controller placement in software defined networks using multi-objective antlion algorithm," *The Journal of Supercomputing*, vol. 78, no. 4, pp.5626-5649, 2022.
- [12] Y. Wu, S. Zhou, Y. Wei, and S. Leng, "Deep reinforcement learning for controller placement in software defined network," in *Proc. IEEE Conf. on Computer Communications Workshops*, pp.1254-1259, Toronto, Canada, 6-9 Jul. 2020.
- [13] H. Mostafaei, M. Menth, and M. S. Obaidat, "A learning automaton-based controller placement algorithm for software-defined networks," in *Proc. IEEE Global Communications Conf.*, 6 pp., Abu Dhabi, United Arab Emirates, 9-13 Dec. 2018.
- [14] S. S. Nande, O. Lhamo, S. Biswas, R. Bassoli, and F. H. Fitzek, "Quantum machine learning for controller placement in software defined networks," in *Proc. 28th European Wireless Conf.*, pp.382-387, Rome, Italy, 2-4 Oct. 2023.
- [15] S. Rahman, *et al.*, "Virtualized controller placement for multi-domain optical transport networks using machine learning," *Photonic Network Communications*, vol. 40, pp.126-136, 2020.
- [16] V. Srivastava and R. S. Pandey, "Machine intelligence approach: To solve load balancing problem with high quality of service performance for multi controller based software defined network," *Sustainable Computing: Informatics and Systems*, vol. 30, Article ID: 100511, 2021.
- [17] C. Kurra, V. P. Dadi, A. Awasthi, V. Janyani and R. B. Battula, "CAPFUL - Coherent Dynamic Controller Placement Method using deep reinforcement learning for SDN," in *Proc. Int. Conf. on Communication, Computing, Networking, and Control in Cyber-*