چکیده

با افزایش ترافیک چندرسانه‌ای و نیاز روزافزون به کیفیت بالای سرویس در شبکه‌های امروزی، روش‌های سنتی مسیریابی قادر به پاسخگویی مؤثر به شرایط متغیر شبکه نیستند. معماری شبکه‌های نرم‌افزارمحور[[1]](#footnote-1) (SDN) با جداسازی صفحه کنترل و داده، امکان مدیریت متمرکز را فراهم کرده اما الگوریتم‌های مسیریابی موجود در کنترلرها اغلب تنها به معیارهای ساده‌ای همچون تعداد پرش متکی هستند. این موضوع در شرایط پرترافیک منجر به افزایش تأخیر و افت کارایی می‌شود.

در این پژوهش، رویکردی مبتنی بر یادگیری ماشین و رگرسیون لجستیک برای پیش‌بینی سطح ترافیک و انتخاب مسیر بهینه ارائه شده است. شبیه‌سازی‌ها در محیط NS2 با زبان C++ و با بهره‌گیری از Floodlight SDN Controller انجام شده است. برای ارزیابی کارایی روش پیشنهادی، دو توپولوژی متفاوت شامل شبکه‌ای با ۹ سوئیچ و شبکه‌ای با ۱۳ سوئیچ طراحی گردید و نتایج در سناریوهای گوناگون با روش پایه مقایسه شد.

شاخص‌های ارزیابی شامل تأخیر متوسط[[2]](#footnote-2)، نرخ تحویل بسته‌ها[[3]](#footnote-3) (PDR)، گذردهی[[4]](#footnote-4) ، نرخ پردازش تراکنش‌ها[[5]](#footnote-5) (TPR) و میزان استفاده از منابع[[6]](#footnote-6) مورد بررسی قرار گرفت. نتایج نشان داد که روش پیشنهادی نسبت به روش پایه بهبود چشمگیری ایجاد کرده است؛ به گونه‌ای که در هر دو توپولوژی، تأخیر کاهش یافته، نرخ تحویل بسته‌ها و گذردهی افزایش یافته و کارایی کلی شبکه ارتقا یافته است.

به‌طور کلی، یافته‌های این تحقیق نشان می‌دهد که به‌کارگیری الگوریتم‌های یادگیری ماشین در کنار معماری SDN می‌تواند راهکاری کارآمد برای مدیریت پویا و بهینه‌سازی ترافیک چندرسانه‌ای در شبکه‌های آینده باشد.

1- مقدمه

رشد سریع فناوری‌های ارتباطی و افزایش تصاعدی تعداد دستگاه‌های متصل به شبکه، منجر به تغییرات بنیادین در ساختار و کارکرد شبکه‌های کامپیوتری شده است. بنا بر گزارش انجام شده، محتوای ویدئویی با سهمی نزدیک به ۳۸ درصد از کل ترافیک اینترنت در شبکه‌های ثابت و سیار، بیشترین حجم انتقال داده را به خود اختصاص داده است. این غلبه نشان‌دهنده‌ی اهمیت و تأثیر عمیق داده‌های چندرسانه‌ای بر زیرساخت‌های جهانی اینترنت است. در نتیجه، افزایش چشمگیر حجم ترافیک و نیاز روزافزون به سرویس‌هایی با کیفیت بالا (نظیر استریم ویدئو، بازی‌های برخط و کنفرانس‌های ویدئویی) به چالشی اساسی در مدیریت شبکه تبدیل شده است.

شبکه‌های سنتی، به دلیل ماهیت غیرمتمرکز مدیریت، عدم استانداردسازی بین تجهیزات و زیرساخت‌های پیچیده، توانایی کافی برای پاسخگویی به این شرایط را ندارند. در این معماری‌ها، تصمیم‌گیری‌های مسیریابی توسط هر گره‌ی شبکه به‌صورت مستقل و محلی انجام می‌شود، بدون آنکه دیدگاه جامع و یکپارچه‌ای از کل شبکه وجود داشته باشد. این ساختار علاوه بر افزایش تأخیر و کاهش کارایی، ادغام قابلیت‌های جدید را دشوار کرده و هزینه‌های مدیریت را به‌طور قابل ملاحظه‌ای بالا می‌برد.

در سال ۲۰۰۸، مفهوم شبکه‌های نرم‌افزارمحور (SDN) در دانشگاه استنفورد برای رفع همین مشکلات معرفی شد . SDN با جداسازی صفحه‌ی کنترل از صفحه‌ی داده و تمرکز تصمیم‌گیری در یک کنترلر مرکزی مانند (Floodlight یا Ryu) تحولی اساسی در مدیریت شبکه ایجاد کرد. در این معماری، کنترلر SDN که به‌مثابه‌ی «مغز شبکه» عمل می‌کند، وظیفه‌ی اعمال قوانین مسیریابی را بر عهده دارد؛ در حالی‌که سوئیچ‌ها و روترها در صفحه‌ی داده صرفاً به انتقال بسته‌ها بر اساس این قوانین می‌پردازند. این رویکرد متمرکز، قابل برنامه‌ریزی و انعطاف‌پذیر، امکان اعمال سیاست‌های جامع‌تر، مدیریت پویا و بهینه‌سازی کارایی را فراهم می‌آورد.

با وجود این مزایا، کنترل‌کننده‌های SDN موجود اغلب برای تصمیم‌گیری در مورد مسیرها به معیارهای ساده‌ای همچون تعداد پرش[[7]](#footnote-7) یا تأخیر متوسط متکی هستند. این امر در کاربردهایی با نیاز به پهنای باند بالا و حساسیت به تأخیر (مانند استریم ویدئو) ناکافی است و در عمل منجر به افت کیفیت تجربه‌ی کاربر می‌شود. بنابراین، توسعه‌ی الگوریتم‌های پیشرفته‌تر که بتوانند شرایط شبکه را به‌صورت پویا و در زمان واقعی ارزیابی کرده و مسیرهای بهینه‌تری انتخاب کنند، ضرورتی انکارناپذیر است.

پیشرفت‌های چشمگیر در حوزه‌ی هوش مصنوعی[[8]](#footnote-8) (AI) و به‌ویژه یادگیری ماشین[[9]](#footnote-9) (ML)، فرصت‌های تازه‌ای برای بهینه‌سازی عملکرد شبکه ایجاد کرده‌اند. الگوریتم‌های هوش مصنوعی قادرند با تجزیه ‌و تحلیل داده‌های ترافیکی، تشخیص ناهنجاری‌ها، پیش‌بینی ازدحام، و انتخاب مسیرهای بهینه، کارایی شبکه را ارتقا دهند. در این میان، استفاده از مدل‌های یادگیری ماشین سبک مانند رگرسیون لجستیک اهمیت ویژه‌ای دارد؛ چرا که این مدل‌ها علاوه بر دقت بالا، هزینه‌ی محاسباتی کمی داشته و برای استقرار بلادرنگ در کنترلرها یا حتی تجهیزات لبه‌ای مناسب‌اند.

1-1 انگیزه و اهداف مطالعه

انگیزه‌ی اصلی این پژوهش، غلبه بر محدودیت‌های کنترل‌کننده‌های SDN موجود است که صرفاً بر معیارهای ابتدایی مسیریابی متکی‌اند. هدف ما توسعه‌ی یک الگوریتم هوشمند، پویا و چندمعیاره است که بتواند ترافیک شبکه را به شکل تطبیقی مدیریت کرده و نیازهای کاربردهای پرمصرف و حساس به کیفیت سرویس[[10]](#footnote-10) را پاسخ دهد.

1-2 مشارکت‌های کلیدی این پژوهش

مطالعه‌ی حاضر چندین دستاورد مهم ارائه می‌دهد:

* ارائه‌ی یک الگوریتم مسیریابی مبتنی بر هوش مصنوعی در چارچوب SDN که مسیرهای انتقال داده را بر اساس شرایط واقعی و لحظه‌ای شبکه انتخاب می‌کند.
* استفاده از NS2 و زبان C++ برای شبیه‌سازی دقیق در سطح بسته و جمع‌آوری داده‌های ترافیکی، در کنار بهره‌گیری از Floodlight SDN Controller برای مدیریت جریان‌ها.
* طراحی و بررسی دو توپولوژی متفاوت (۹ و ۱۳ سوئیچ) با سناریوهای ترافیکی متنوع به منظور ارزیابی جامع الگوریتم پیشنهادی.
* ایجاد یک مجموعه‌داده اختصاصی برای طبقه‌بندی سطوح ترافیک و آموزش مدل‌های مختلف یادگیری ماشین. در این میان، رگرسیون لجستیک به‌عنوان بهترین مدل انتخاب شده و در الگوریتم نهایی ادغام شده است.
* ارزیابی عملکرد الگوریتم پیشنهادی در مقایسه با روش‌های پایه و آگاه به کیفیت سرویس[[11]](#footnote-11) موجود، بر اساس معیارهایی چون تأخیر متوسط ، نرخ تحویل بسته‌ها (PDR)، گذردهی، نرخ پردازش تراکنش‌ها (TPR) و میزان استفاده از منابع.
* نتایج نشان می‌دهد که الگوریتم پیشنهادی باعث کاهش محسوس تأخیر، افزایش گذردهی و بهبود کیفیت ویدئو نسبت به روش‌های سنتی شده است.
* در نهایت، پژوهش حاضر نشان می‌دهد که ادغام هوش مصنوعی در چارچوب SDN می‌تواند راهکاری انعطاف‌پذیر، کارآمد و کاربرمحور برای مدیریت ترافیک در برنامه‌های با تقاضای بالا ارائه دهد.

بخش 2 به مرور کارهای مرتبط و وضعیت موجود اختصاص دارد. در بخش 3 روش پیشنهادی معرفی می‌شود. سپس در بخش 4 جزئیات ارزیابی عملکرد ارائه شده و در بخش 5 به بحث و تحلیل نتایج پرداخته می‌شود. در نهایت، جمع‌بندی مطالب در بخش 6 آورده شده است.

2- کارهای مرتبط و وضعیت موجود

یکی از تلاش‌ها در زمینه‌ی بهبود مسیریابی و توزیع بار در شبکه‌های نرم‌افزارمحور توسط ورساما صورت گرفت که در آن الگوریتم کلاسیک Equal-Cost Multipath (ECMP) و نسخه‌ی ارتقاءیافته‌ی آن مبتنی بر متعادل‌سازی بار پهنای‌باند مورد بررسی قرار گرفت. در حالی که ECMP جریان‌ها را به‌طور ایستا و بدون درنظرگرفتن وضعیت کلی شبکه توزیع می‌کرد، نسخه‌ی اصلاح‌شده با اتکا به آمارهای کنترلی توانست بهبود قابل‌توجهی در گذردهی ایجاد کند. با این حال، آزمایش‌های محدود بر روی توپولوژی کوچک و نادیده گرفتن سربار پردازنده‌ی کنترلر، استفاده از این رویکرد در مقیاس‌های بزرگ‌تر را با تردید همراه ساخت. در ادامه، یولو و همکاران با تمرکز بر مراکز داده، نشان دادند که بهره‌گیری از الگوریتم‌های یادگیری ماشین نظیر ANN، SVM و Logistic Regression می‌تواند منجر به توزیع بار پویا و بهبود عملکرد کلی سیستم شود. هرچند نتایج اولیه مثبت بود، اما استفاده از داده‌های محدود و ضعف در تبیین‌پذیری مدل‌های پیچیده همچون ANN، مانع از تعمیم‌پذیری این یافته‌ها به محیط‌های واقعی شد.

از سوی دیگر، الهلالی و همکاران با مرور جامع پژوهش‌های پس از سال ۲۰۱۷، متعادل‌کننده‌های بار را در چهار دسته‌ی الهام‌گرفته از طبیعت، مبتنی بر یادگیری ماشین، مبتنی بر مدل‌های ریاضی و سایر روش‌ها طبقه‌بندی کرده و به‌روشنی نشان دادند که بخش بزرگی از آثار گذشته تنها به زیرمجموعه‌ای از پارامترهای QoS توجه کرده‌اند. این محدودیت سبب شد تا تحقیقات بعدی بر استفاده‌ی همزمان از مجموعه‌ی کامل‌تری از شاخص‌ها متمرکز شوند. در همین راستا، حامد و همکاران با تمرکز بر مراکز داده و استفاده از SDN به‌عنوان جایگزینی برای متعادل‌کننده‌های بار سنتی، رویکردی مبتنی بر پهنای‌باند را پیشنهاد کردند که در مقایسه با روش‌های Round Robin و Links عملکرد بهتری داشت. با این حال، آزمایش‌های محدود در مقیاس کوچک و کاهش کارایی پس از رسیدن به آستانه‌ی مشخصی از بار شبکه، پرسش‌هایی را درخصوص مقیاس‌پذیری این روش برانگیخت.

تلاش‌های دیگر بیشتر بر استفاده از روش‌های فرا ابتکاری متمرکز بود. به‌عنوان نمونه، بلگام و همکاران دو الگوریتم بهینه‌سازی کلونی مورچه‌ها[[12]](#footnote-12) (ACO) و ازدحام ذرات[[13]](#footnote-13) (PSO) را برای متعادل‌سازی بار در شبکه‌های SDN مقایسه کردند و نشان دادند که PSO تأخیر کمتری ایجاد می‌کند. با وجود این، استفاده از چنین الگوریتم‌هایی به دلیل زمان همگرایی بالا و وابستگی به انتخاب مناسب پارامترها، برای محیط‌های زمان‌واقعی چندان مناسب به‌نظر نمی‌رسد. در مطالعه‌ای مشابه، دگیرمنچی و همکاران الگوریتم‌های جستجوی خطی، جستجوی تابو و آنیلینگ شبیه‌سازی‌شده را ترکیب کردند و گرچه نتایج نشان دادند که روش‌های ترکیبی در برخی شرایط عملکرد بهتری دارند، اما همچنان عدم تضمین بهینگی و هزینه‌ی بالای محاسباتی از محدودیت‌های این دسته از رویکردها به‌شمار می‌رود.

با توجه به اهمیت سرویس‌های چندرسانه‌ای، پژوهش‌هایی نیز به‌طور خاص به بهبود کیفیت تجربه‌ی کاربر[[14]](#footnote-14) (QoE) پرداخته‌اند. کوک و همکاران با تمرکز بر محتوای ویدئویی در شبکه‌های 5G، الگوریتمی مبتنی بر پروتکل دسترسی کاربر پیشنهاد دادند و نشان دادند که انتخاب مسیر بر اساس تراکم ترافیک، نسبت به روش‌های مبتنی بر DNS و تعداد پرش، تا ۶۰٪ بهبود در کیفیت ویدئو ایجاد می‌کند. هرچند این نتایج امیدوارکننده بود، اما نبود مدیریت کش در CDN و عدم توجه به پویایی مسیرها، محدودیت جدی این رویکرد بود. در حوزه‌ی شبکه‌های توزیع محتوا[[15]](#footnote-15) (CDN)، بوروس و همکاران با ترکیب الگوریتم‌های ML از جمله Decision Tree و Random Forest، نشان دادند که می‌توان بار را به‌طور متوازن بین سرورها توزیع کرد و به دقتی نزدیک به ۹۷٪ در پیش‌بینی مسیر دست یافت. با این وجود، تمرکز بیش‌ازحد بر دقت مدل و بی‌توجهی به سربار جمع‌آوری داده‌ها و استنتاج بلادرنگ، قابلیت عملیاتی این رویکرد را محدود ساخت.

از منظر امنیت، ارسوی و همکاران تهدیدات امنیتی در SDN را با بهره‌گیری از الگوریتم‌های یادگیری ماشین بررسی کردند و با ترکیب مجموعه‌داده‌ها و اعمال الگوریتم‌هایی چون Random Forest و XGBoost، توانستند نرخ شناسایی تهدید بالایی به‌دست آورند. با این حال، محدودیت محیط آزمایشگاهی و ترکیب مصنوعی داده‌ها، نتایج را به‌شدت وابسته به سناریوهای خاص ساخت. مرورهای جامع‌تری مانند کار عثمان و همکاران نیز نشان دادند که اگرچه ادغام AI در SDN و NFV فرصت‌های بزرگی در مدیریت منابع، تشخیص خطا و امنیت ایجاد می‌کند، اما در عمل پژوهش‌های کاربردی و میدانی اندک است و چالش‌های جدی مقیاس‌پذیری و امنیتی همچنان پابرجاست.

در مجموع، مرور ادبیات نشان می‌دهد که مسیر حرکت تحقیقات از روش‌های ایستا و تک‌معیاره به سمت استفاده از الگوریتم‌های پویا، چندمعیاره و مبتنی بر یادگیری ماشین بوده است. با این حال، بیشتر مطالعات موجود داده‌های محدودی داشته‌اند یا تنها بر دقت مدل متمرکز بوده و سربار محاسباتی و عملیاتی را نادیده گرفته‌اند.

پژوهش حاضر دقیقاً در همین نقطه جای می‌گیرد: با بهره‌گیری از NS2 و زبان C++، در کنار کنترلر Floodlight، یک الگوریتم مبتنی بر رگرسیون لجستیک ارائه می‌شود که ضمن برخورداری از هزینه‌ی محاسباتی پایین و قابلیت استقرار زمان‌واقعی، بر پایه‌ی مجموعه‌ای چندمعیاره از شاخص‌های QoS تصمیم‌گیری می‌کند. بدین ترتیب، برخلاف بسیاری از آثار پیشین که صرفاً به اثبات مفهوم در محیط‌های ساده بسنده کرده‌اند، این مطالعه گامی به سمت کاربردپذیری عملی و امکان استقرار در سناریوهای پیچیده‌تر برداشته است.

2-1 جمع‌بندی پژوهش‌های مرتبط

از مرور مطالعات فوق چند نکته کلیدی به دست می‌آید:

* الگوریتم‌های سنتی ساده‌اند اما به دلیل عدم درک شرایط لحظه‌ای شبکه ناکارآمد هستند.
* الگوریتم‌های هوش مصنوعی (ANN، SVM و ...) توانسته‌اند پیشرفت‌های محسوسی در QoS ایجاد کنند، ولی بیشتر روی توپولوژی‌های کوچک آزمایش شده‌اند.
* بیشتر تحقیقات در Mininet انجام شده و محیط‌های دقیق‌تر مانند NS2 کمتر مورد استفاده قرار گرفته‌اند.
* ترکیب SDN + AI آینده‌ی مسیریابی پویا و مدیریت شبکه است، اما نیاز به کارهای بیشتر در مقیاس بزرگ و شرایط واقعی دارد.

2-2 جایگاه پژوهش حاضر

در این پژوهش، الگوریتم AIRSDN[[16]](#footnote-16) در محیط NS2 پیاده‌سازی شده است. تفاوت اصلی کار ما با مطالعات پیشین این است که:

* به‌جای تکیه بر Mininet، از NS2 استفاده شده که معیارهای دقیق‌تری ارائه می‌دهد.
* به‌جای تمرکز صرف بر توان عملیاتی یا تأخیر، پنج معیار کلیدی Throughput، Delay، PDR، Usage، TPR به‌طور همزمان ارزیابی شده‌اند.
* الگوریتم پیشنهادی نه‌تنها کیفیت سرویس را بهبود داده، بلکه کارایی منابع و نرخ پردازش تراکنش را هم ارتقا داده است.

3- روش پیشنهادی

شبکه‌های نرم‌افزارمحور (SDN) بر اساس معماری سه‌لایه متشکل از لایه داده، لایه کنترل و لایه برنامه تعریف می‌شوند. ارتباط میان لایه داده و کنترل از طریق رابط جنوبی[[17]](#footnote-17) صورت می‌گیرد که رایج‌ترین پروتکل در آن OpenFlow است. در این چارچوب، هرگاه بسته‌ای وارد سوئیچ شود، ابتدا جدول جریان آن بررسی می‌گردد. اگر قانون متناظری وجود نداشته باشد، سوئیچ درخواست تصمیم را به کنترلر ارسال می‌کند. کنترلر بر اساس وضعیت شبکه و سیاست‌های مسیریابی، قاعده مناسب را تولید و جدول جریان سوئیچ را به‌روزرسانی می‌کند. این سازوکار که بر الگوی تطبیق–عمل[[18]](#footnote-18) استوار است، مبنای مدیریت پویای مسیرها در SDN محسوب می‌شود. در سطح بالاتر، رابط شمالی[[19]](#footnote-19) ارتباط میان لایه کنترل و لایه برنامه را فراهم می‌آورد و امکان مشاهده توپولوژی شبکه، پایش تجهیزات و تعریف سیاست‌های جدید را به توسعه‌دهندگان می‌دهد.

در این پژوهش، یک الگوریتم مسیریابی تطبیقی چندمعیاره مبتنی بر رگرسیون لجستیک ارائه شده است. هدف اصلی این الگوریتم بهبود فرآیند انتخاب مسیر با در نظر گرفتن مجموعه‌ای از شاخص‌های عملکردی شبکه است، به‌گونه‌ای که محدودیت رویکردهای سنتی مبتنی بر معیارهای ساده‌ای همچون hop count برطرف شود.

3-1 چرخه عملکرد الگوریتم پیشنهادی

مراحل اجرای الگوریتم به شرح زیر است:

1. آماده‌سازی مدل: پیش از آغاز انتقال داده، مدل رگرسیون لجستیک که از پیش با داده‌های شبکه آموزش دیده است، در کنترلر Floodlight بارگذاری می‌شود. این مدل وظیفه دارد سطح بار مسیرهای مختلف را پیش‌بینی نماید.
2. استخراج مسیرهای کاندید: کنترلر Floodlight چند مسیر کوتاه میان کلاینت و سرور شناسایی می‌کند و این مسیرها از طریق ماژول StaticEntryPusher در اختیار الگوریتم قرار می‌گیرند.
3. جمع‌آوری داده‌های جریان: برای هر مسیر انتخاب‌شده، قوانین موقت در جدول جریان[[20]](#footnote-20) سوئیچ‌ها نصب می‌شوند تا آمار جریان‌ها شامل تأخیر (Delay)، نرخ تحویل بسته‌ها (PDR)، گذردهی (Throughput)، نرخ پردازش تراکنش‌ها (TPR) و نرخ استفاده از منابع (Usage) گردآوری شود.
4. پیش‌بینی سطح ترافیک: داده‌های جمع‌آوری‌شده به مدل رگرسیون لجستیک داده می‌شوند تا مسیرها از نظر سطح بار به «کم‌ترافیک» یا «پرترافیک» طبقه‌بندی شوند.
5. انتخاب مسیر بهینه: مسیری که کمترین سطح ازدحام را دارد انتخاب و قوانین پایدار در سوئیچ‌های مربوطه نصب می‌گردند تا انتقال داده (مانند جریان ویدئو) آغاز شود.
6. بازپیکربندی پویا: الگوریتم در بازه‌های زمانی از پیش تعریف‌شده مجدداً اجرا می‌شود تا در صورت تغییر شرایط ترافیکی، مسیر مناسب‌تری انتخاب گردد. این ویژگی پایداری کیفیت و سازگاری روش با شرایط متغیر شبکه را تضمین می‌کند.

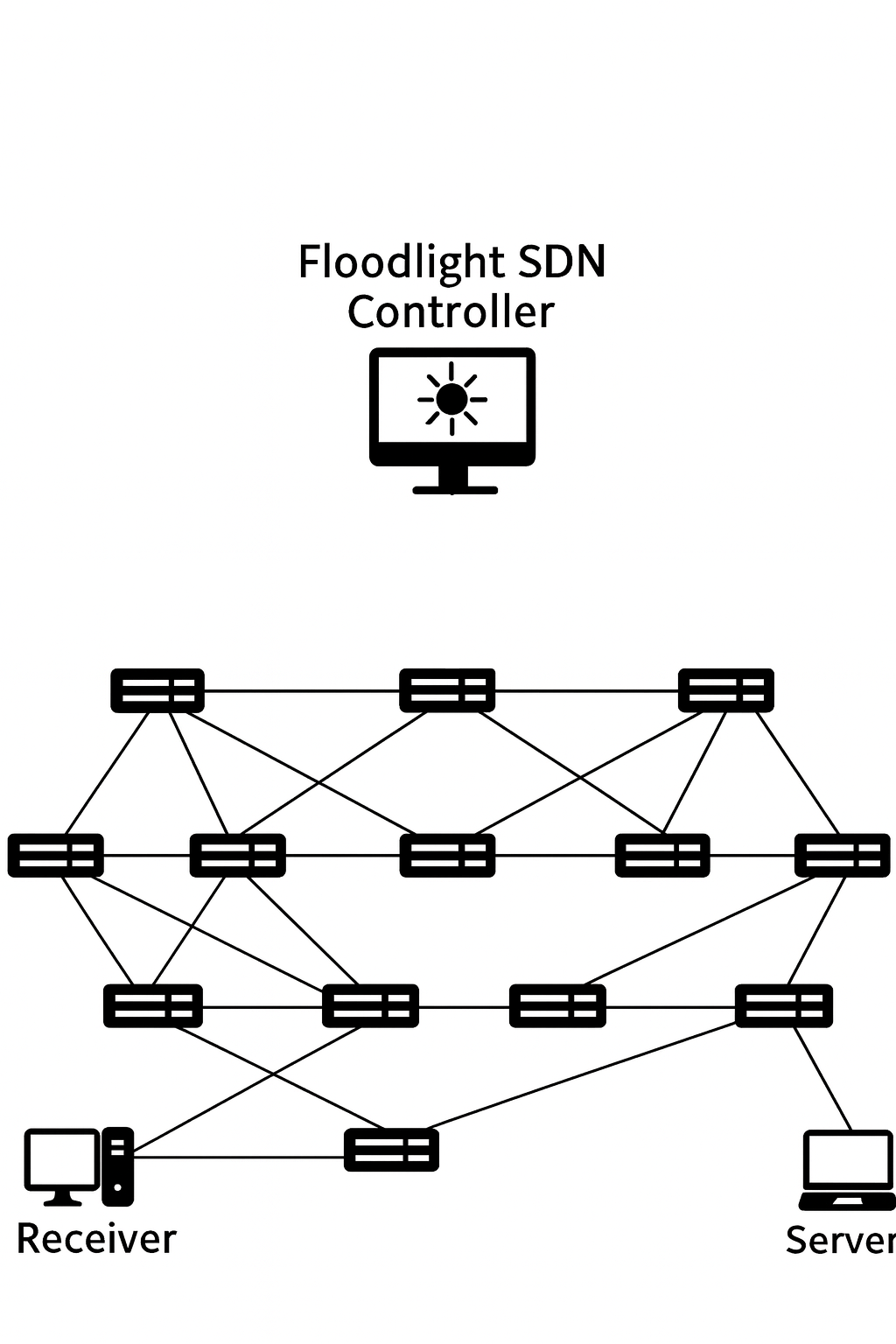
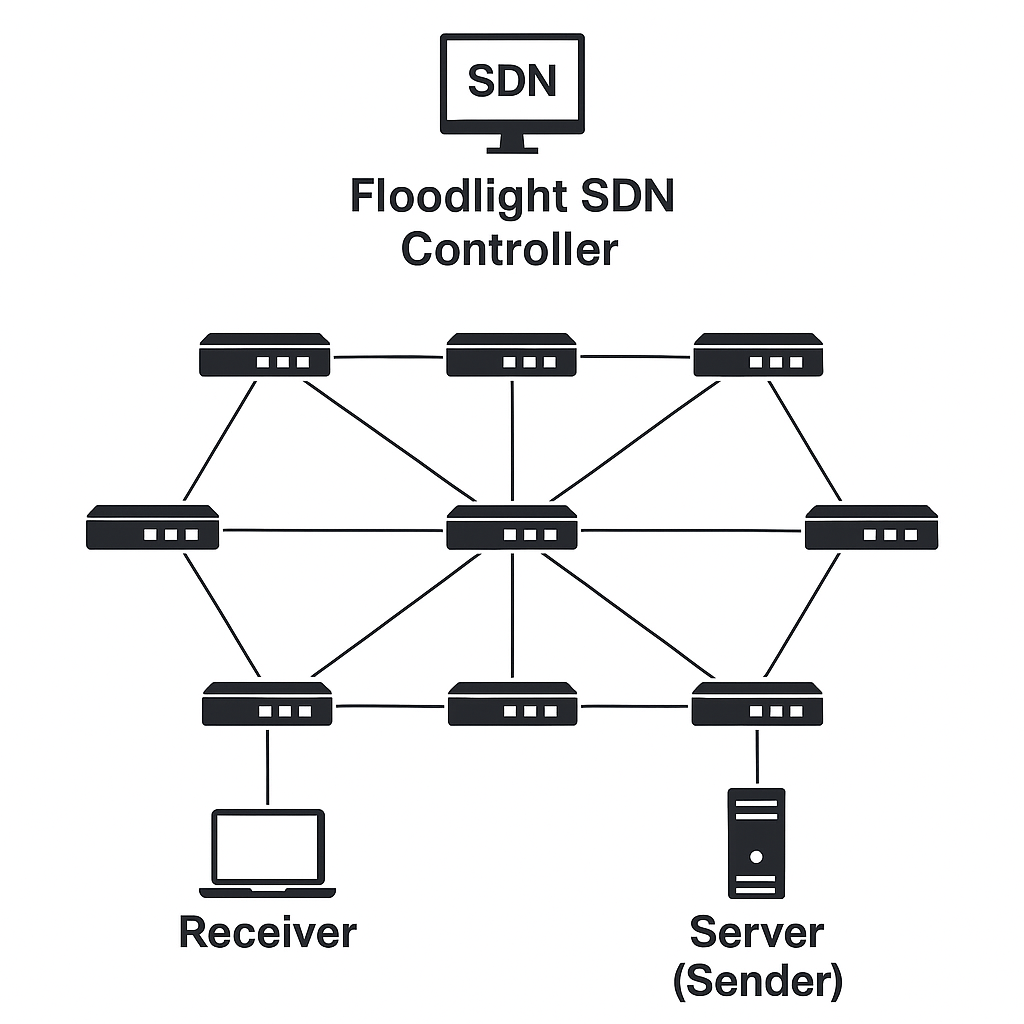
نکته‌ی کلیدی در این رویکرد استفاده از رگرسیون لجستیک به‌عنوان مدل طبقه‌بندی‌کننده است. برخلاف الگوریتم‌های پیچیده‌تری مانند شبکه‌های عصبی عمیق، این مدل دارای هزینه‌ی محاسباتی بسیار پایین بوده و به راحتی در محیط کنترلرهای SDN قابل پیاده‌سازی است. علاوه بر این، قابلیت تفسیرپذیری بالای آن امکان تحلیل دقیق‌تر تأثیر هر ویژگی بر سطح ترافیک را فراهم می‌کند. این ویژگی در محیط‌های شبکه‌ای که نیازمند تصمیم‌گیری سریع و شفاف هستند، مزیتی چشمگیر به‌شمار می‌رود.

3-2 محیط شبیه‌سازی و سناریوها

برای اعتبارسنجی روش پیشنهادی، شبیه‌سازی در NS2 با زبان C++ انجام شده و کنترلر Floodlight به‌عنوان هسته مدیریتی شبکه استفاده شده است. همچنین برای تولید داده‌های آموزشی و تکمیلی، از NS2 استفاده شده است.

در این مطالعه، دو توپولوژی متفاوت به‌عنوان سناریوی آزمایش در نظر گرفته شدند:

* توپولوژی سناریو اول: شامل ۹ سوئیچ، یک کلاینت و یک سرور.
* توپولوژی سناریو دوم: شامل ۱۳ سوئیچ، یک کلاینت و یک سرور.



شکل 2- سناریو دوم (13 سوئیچ)

شکل 1- سناریو اول (9 سوئیچ)

در هر دو توپولوژی، بار شبکه با استفاده از ابزار iperf3 و در قالب اتصال‌های TCP تولید گردید. انتخاب TCP به دلیل قابلیت آن در ارائه انتقال داده قابل اعتماد و مدیریت ازدحام انجام شد. برای هر توپولوژی دو سطح بار تعریف شد:

* کم‌ترافیک: چهار کلاینت با دو اتصال TCP حدود ۳۰٪ پهنای‌باند را مصرف کردند.
* پرترافیک: هشت کلاینت با چهار اتصال TCP نزدیک به ۹۰٪ پهنای‌باند را مصرف نمودند.

برای افزایش واقع‌گرایی، در هر سناریو انتقال ویدئو بین کلاینت و سرور انجام گرفت. گردآوری داده‌ها و اجرای آزمایش‌ها به‌صورت خودکار با استفاده از اسکریپت‌های C++ انجام گردید. وظایف مختلف مانند تولید بار، ثبت آمار، استنتاج مدل و نصب قوانین جریان نیز به‌صورت چندریسمانی[[21]](#footnote-21) اجرا شدند تا تأخیر تصمیم‌گیری در کنترلر به حداقل برسد.

3-3 الگوریتم روش پیشنهادی:

Algorithm 1: AI-based Adaptive Routing in SDN (Logistic Regression)

Input:

G(V,E) : Graph of network topology (switches and links)

S, D : Source and Destination nodes

k : Number of candidate paths (k-shortest paths)

Δt : Update interval for re-evaluation

ML\_Model : Pre-trained Logistic Regression model

Output:

P\* : Selected optimal path between S and D

Begin

1. Initialize Floodlight Controller

2. while (Transmission ongoing) do

3. CandidatePaths ← Get K ShortestPaths(G, S, D, k)

4. For each path p ∈ CandidatePaths do

5. Install temporary flow rules for p

6. Collect flow statistics:

- Delay(p)

- Throughput(p)

- Packet Delivery Ratio (PDR(p))

- Transaction Processing Rate (TPR(p))

- Resource Usage (Usage(p))

7. FeatureVector ← {Delay(p), Throughput(p), PDR(p), TPR(p), Usage(p)}

8. PredictedTrafficLevel(p) ← Model.predict(FeatureVector)

9. End For

10. P\* ← argmin {PredictedTrafficLevel(p)} for all p ∈ CandidatePaths

11. Install stable flow rules for P\*

12. Wait Δt

13. End while

End

3-3-1 تشریح فنی الگوریتم پیشنهادی

الگوریتم مسیریابی تطبیقی پیشنهادی بر پایه‌ی رگرسیون لجستیک طراحی شده و در محیط SDN با کنترلر Floodlight پیاده‌سازی گردیده است. روند کار الگوریتم مطابق شبه‌کد (Algorithm 1) در چند مرحله‌ی اصلی انجام می‌شود که در ادامه به‌صورت فنی تشریح می‌گردد.

1. مقداردهی اولیه کنترلر

در ابتدای کار، کنترلر Floodlight فعال شده و ارتباط آن با سوئیچ‌ها از طریق پروتکل OpenFlow برقرار می‌شود. اطلاعات توپولوژی شبکه شامل مجموعه سوئیچ‌ها و لینک‌ها استخراج و به‌صورت گراف G(V,E) مدل‌سازی می‌شود. در این مرحله، تنظیمات مربوط به زمان‌انقضا (Idle/Hard Timeout) قوانین جریان نیز تعریف می‌شوند تا از انباشت قوانین قدیمی جلوگیری شود.

2. اجرای حلقه‌ی انطباقی

از آنجا که وضعیت شبکه پویا و متغیر است، الگوریتم در قالب یک حلقه‌ی تکرارشونده اجرا می‌شود. این حلقه تا پایان انتقال داده فعال بوده و در هر بازه‌ی زمانی Δt مجدداً شرایط شبکه را بررسی کرده و در صورت نیاز مسیر جایگزین انتخاب می‌کند.

3. استخراج مسیرهای کاندید

کنترلر Floodlight با استفاده از الگوریتم‌های یافتن چند مسیر کوتاه[[22]](#footnote-22) مجموعه‌ای از مسیرهای بالقوه بین مبدأ و مقصد را استخراج می‌کند. انتخاب چند مسیر به‌عنوان ورودی الگوریتم، انعطاف‌پذیری بیشتری در فرآیند تصمیم‌گیری ایجاد می‌نماید.

4. نصب قواعد موقت و جمع‌آوری داده‌ها

برای هر مسیر کاندید، قوانین موقت در جدول جریان سوئیچ‌ها نصب می‌شود. سپس، آمار جریان‌ها طی یک پنجره‌ی زمانی مشخص جمع‌آوری می‌گردد. داده‌های گردآوری‌شده شامل شاخص‌های کلیدی زیر هستند:

* تأخیر (Delay)
* نرخ تحویل بسته‌ها (PDR)
* گذردهی (Throughput)
* نرخ پردازش تراکنش‌ها (TPR)
* میزان استفاده از منابع (Usage)

به‌منظور کاهش سربار، قوانین موقت با اولویت پایین و زمان‌انقضای کوتاه نصب می‌شوند تا پس از پایان اندازه‌گیری به‌طور خودکار حذف گردند.

5. تشکیل بردار ویژگی و پیش‌پردازش

آمار به‌دست‌آمده از هر مسیر به‌صورت یک بردار ویژگی p تعریف می‌شود. برای هم‌خوانی با داده‌های آموزشی، مقادیر ویژگی‌ها نرمال‌سازی شده و در صورت وجود نوسان یا داده‌های ناقص، عملیات پیش‌پردازش (میانگین‌گیری، جایگزینی مقادیر گمشده و غیره) انجام می‌شود.

6. پیش‌بینی سطح ترافیک

بردار ویژگی مسیر به مدل رگرسیون لجستیک داده می‌شود. این مدل با استفاده از تابع سیگموید، احتمال ازدحام مسیر را برآورد کرده و مسیرها را در یکی از دو دسته‌ی «کم‌ترافیک» یا «پرترافیک» قرار می‌دهد

7. انتخاب مسیر بهینه

پس از محاسبه‌ی احتمال ازدحام برای تمامی مسیرهای کاندید، مسیری که دارای کمترین مقدار است به‌عنوان مسیر بهینه انتخاب می‌شود. برای جلوگیری از تغییرات مکرر مسیر[[23]](#footnote-23)، مکانیزم هیستریسس اعمال می‌شود. بر این اساس، تنها زمانی مسیر جایگزین انتخاب می‌شود که بهبود احتمال ازدحام از یک آستانه‌ی مشخص بیشتر باشد.

8. نصب قوانین پایدار

پس از انتخاب مسیر بهینه، قوانین پایدار با اولویت بالا در سوئیچ‌های مربوطه نصب می‌شوند و ترافیک اصلی (مانند جریان ویدئو) از آن مسیر هدایت می‌شود.

9. بازپیکربندی دوره‌ای

الگوریتم پس از گذشت بازه‌ی زمانی Δt مجدداً اجرا می‌شود. در صورت تغییر شرایط شبکه، مسیر بهینه به‌روزرسانی شده و قوانین جریان متناسب با آن اصلاح می‌گردند. این بازپیکربندی پویا تضمین می‌کند که انتخاب مسیر همواره منطبق بر شرایط جاری شبکه باشد.

3-4 جمع‌بندی روش پیشنهادی

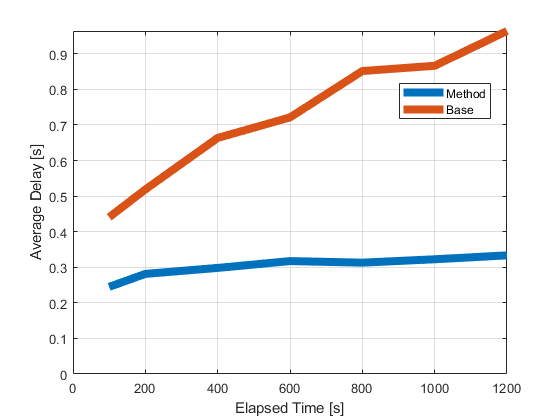
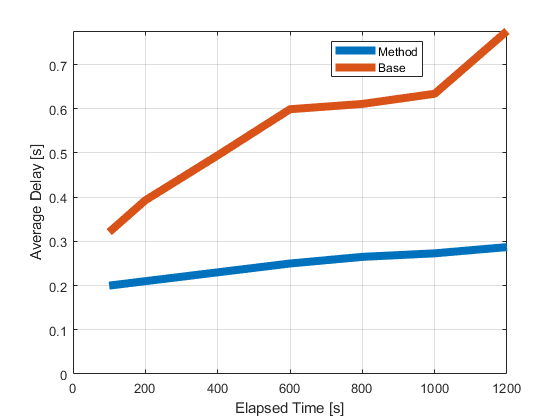
مزیت اصلی روش پیشنهادی در مقایسه با رویکردهای متداول، در تصمیم‌گیری چندمعیاره و پویا نهفته است. در حالی که بسیاری از الگوریتم‌های فعلی تنها به معیارهایی مانند hop count یا پهنای‌باند توجه دارند، روش حاضر مجموعه‌ای غنی از شاخص‌ها را به کار می‌گیرد که شامل تأخیر، گذردهی ، نرخ تحویل بسته‌ها (PDR)، نرخ پردازش تراکنش‌ها (TPR) و استفاده از منابع (Usage) می‌شود. همین چندبعدی بودن ارزیابی مسیرهاست که دقت انتخاب را افزایش داده و منجر به بهبود قابل توجه تجربه‌ی کاربر در انتقال داده‌های حساس، خصوصاً ویدئوهای با کیفیت بالا، می‌گردد.

در مجموع، چارچوب پیشنهادی با ادغام قابلیت‌های نظارتی SDN، توان پیش‌بینی الگوریتم‌های یادگیری ماشین و مکانیسم بازپیکربندی پویا، یک رویکرد کارآمد و سبک برای مسیریابی تطبیقی ارائه می‌دهد. نتایج حاصل از شبیه‌سازی‌ها که در بخش بعدی تشریح می‌شود، نشان می‌دهد که این روش توانسته است ضمن حفظ سربار پایین محاسباتی، کارایی شبکه را به طور معناداری افزایش دهد و کیفیت سرویس را بهبود بخشد.

4- ارزیابی عملکرد

خروجی شبیه‌سازی برای دو سناریوی متفاوت شبکه (توپولوژی ۹ سوئیچی و ۱۳ سوئیچی) بررسی می‌شود. معیارهای کلیدی شامل میانگین تأخیر، گذردهی ، نرخ تحویل بسته‌ها (PDR)، نرخ پردازش تراکنش‌ها (TPR) و استفاده از منابع (Usage) می‌شود. نتایج به‌صورت نمودار ارائه شده و نشان می‌دهند که روش پیشنهادی مبتنی بر رگرسیون لجستیک در مقایسه با روش پایه، در همه‌ی معیارها بهبود قابل توجهی ایجاد کرده است؛ به‌ویژه در کاهش تأخیر و افزایش PDR که تأثیر مستقیم بر کیفیت تجربه‌ی کاربر دارند

1-4 میانگین تاخیر  
نتایج نشان می‌دهند که میانگین تأخیر در روش پایه (Base) در هر دو سناریو با گذشت زمان به‌طور قابل توجهی افزایش می‌یابد و در پایان شبیه‌سازی به حدود ۰٫۹ ثانیه در توپولوژی ۹ سوئیچی و حدود ۰٫۷۵ ثانیه در توپولوژی ۱۳ سوئیچی می‌رسد. این روند افزایشی ناشی از ازدحام و نبود مکانیزم‌های هوشمند توزیع بار است که سبب طولانی‌تر شدن صف‌ها در سوئیچ‌ها می‌شود. در مقابل، روش پیشنهادی (Method) توانسته است در هر دو توپولوژی تأخیر را در سطح پایین و تقریباً پایدار نگه دارد، به‌طوری‌که مقدار میانگین آن بین ۰٫۲ تا ۰٫۳ ثانیه باقی مانده است. مقایسه دو سناریو نشان می‌دهد که با وجود افزایش تعداد سوئیچ‌ها و پیچیدگی مسیرها در توپولوژی بزرگ‌تر، الگوریتم پیشنهادی همچنان کارایی خود را حفظ کرده و از افزایش شدید تأخیر جلوگیری کرده است. بنابراین، استفاده از مدل رگرسیون لجستیک در انتخاب مسیر، باعث کاهش ۵۰ تا ۶۰ درصدی تأخیر نسبت به روش پایه شده و کیفیت تجربه کاربر به‌ویژه در کاربردهای حساس به زمان مانند پخش ویدیو به شکل محسوسی بهبود یافته است.



شکل 3- میانگین تاخیر در سناریو اول

شکل 4- میانگین تاخیر در سناریو دوم

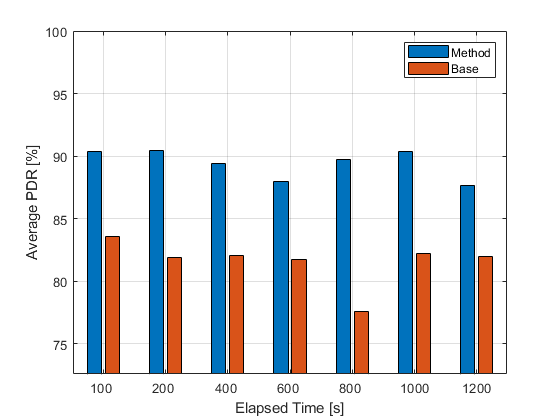
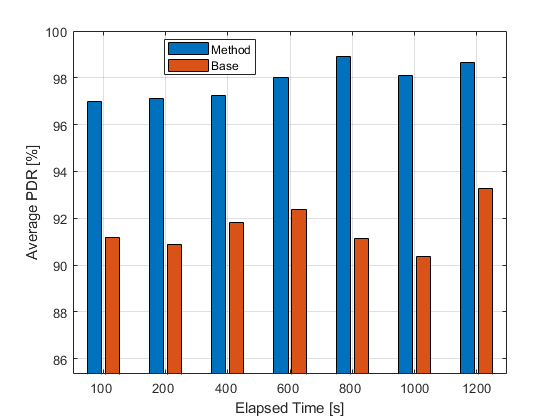
2-4 میانگین نرخ تحویل بسته

نرخ تحویل بسته (PDR) نشان می‌دهد چه نسبتی از بسته‌های ارسال‌شده توسط منبع، بدون خطا به مقصد رسیده‌اند و یکی از مهم‌ترین شاخص‌ها برای ارزیابی قابلیت اطمینان شبکه است.

در سناریوی ۹ سوئیچی، مشاهده می‌شود که روش پایه (Base) نرخ تحویل بسته‌ای در حدود ۷۸٪ تا ۸۴٪ دارد. این مقدار نسبتاً پایین بیانگر آن است که بخشی از بسته‌ها به دلیل ازدحام شبکه یا افت ترافیک از دست می‌روند. در مقابل، روش پیشنهادی (Method) توانسته است PDR را به محدوده‌ی ۸۸٪ تا ۹۱٪ برساند. این بهبود حدود ۸ تا ۱۰ درصدی نشان‌دهنده‌ی توانایی الگوریتم رگرسیون لجستیک در پیش‌بینی وضعیت مسیرها و انتخاب مسیرهای پایدارتر است که منجر به کاهش افت بسته‌ها می‌شود.

در سناریوی ۱۳ سوئیچی، تفاوت میان دو روش حتی بیشتر به چشم می‌خورد. در این حالت، روش پایه مقادیر PDR بین ۹۰٪ تا ۹۳٪ را به ثبت رسانده است، در حالی‌که روش پیشنهادی همواره بالاتر از ۹۷٪ بوده و در برخی زمان‌ها به ۹۹٪ نیز نزدیک شده است. این نتیجه اهمیت استفاده از تصمیم‌گیری چندمعیاره در مسیریابی را برجسته می‌کند، چراکه حتی در توپولوژی‌های پیچیده‌تر نیز افت محسوس بسته‌ها تقریباً حذف شده است.

به‌طور کلی، مقایسه‌ی نتایج دو سناریو نشان می‌دهد که روش پیشنهادی نسبت به روش پایه بین ۵٪ تا ۱۰٪ بهبود در نرخ تحویل بسته‌ها ایجاد کرده است. این بهبود به‌ویژه در کاربردهای حساس مانند پخش ویدئو و تماس‌های بی‌درنگ بسیار حیاتی است، زیرا کاهش افت بسته‌ها مستقیماً به افزایش کیفیت تجربه‌ی کاربر منجر می‌شود.



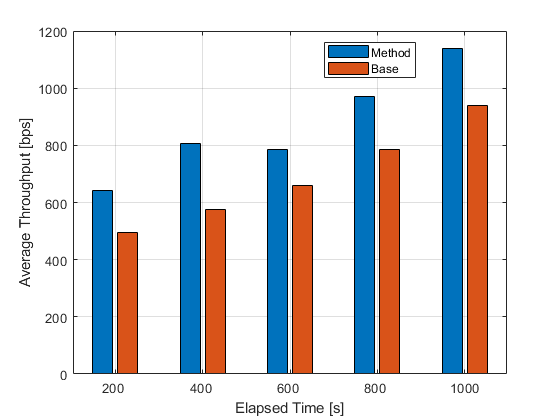
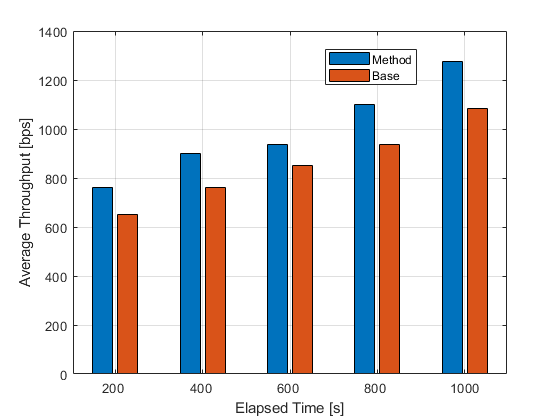
شکل 5- میانگین نرخ تحویل بسته در سناریو اول

شکل 6- میانگین نرخ تحویل بسته در سناریو دوم

3-4 میانگین گذردهی  
گذردهی به عنوان یکی از معیارهای اساسی عملکرد شبکه، بیانگر حجم داده‌هایی است که در یک بازه زمانی مشخص با موفقیت از مبدأ به مقصد منتقل می‌شوند. بررسی نتایج شبیه‌سازی در دو توپولوژی ۹ سوئیچی و ۱۳ سوئیچی نشان می‌دهد که روش پیشنهادی مبتنی بر رگرسیون لجستیک توانسته بهبود قابل توجهی نسبت به روش پایه ارائه دهد. در سناریوی ۹ سوئیچی، روش پایه در زمان ۲۰۰ ثانیه حدود ۵۰۰ بیت بر ثانیه و در پایان شبیه‌سازی حدود ۹۳۰ بیت بر ثانیه گذردهی ثبت کرده است، در حالی‌که روش پیشنهادی در همان نقاط به ترتیب حدود ۶۳۰ و ۱۱۵۰ بیت بر ثانیه را نشان می‌دهد. این نتایج گویای آن است که الگوریتم پیشنهادی با انتخاب مسیرهای کم‌ترافیک و مدیریت بهتر منابع، توانسته میانگین گذردهی را به طور متوسط ۲۰ درصد افزایش دهد.

در توپولوژی ۱۳ سوئیچی، تفاوت عملکرد دو روش بیشتر آشکار می‌شود. در این سناریو روش پایه در بهترین حالت حدود ۱۱۰۰ بیت بر ثانیه گذردهی داشته، در حالی‌که روش پیشنهادی در همان شرایط توانسته مقدار ۱۲۸۰ بیت بر ثانیه را ثبت کند. همچنین در بازه‌های میانی مانند ۶۰۰ ثانیه، گذردهی روش پایه حدود ۸۶۰ بیت بر ثانیه و گذردهی روش پیشنهادی حدود ۹۵۰ بیت بر ثانیه بوده است. این اختلاف نشان می‌دهد که هرچه پیچیدگی شبکه افزایش یابد، قدرت الگوریتم پیشنهادی در یافتن مسیرهای بهینه و جلوگیری از ایجاد گلوگاه‌ها بیشتر نمایان می‌شود.

در مجموع می‌توان نتیجه گرفت که استفاده از رویکرد پیشنهادی منجر به افزایش میانگین گذردهی شبکه در هر دو توپولوژی شده و این بهبود، که بین ۱۵ تا ۲۵ درصد متغیر است، مستقیماً حاصل تصمیم‌گیری چندمعیاره و قابلیت بازپیکربندی پویا مسیرها در شرایط پرترافیک شبکه است.



شکل 8- میانگین گذردهی در سناریو دوم

شکل 7- میانگین گذردهی در سناریو اول

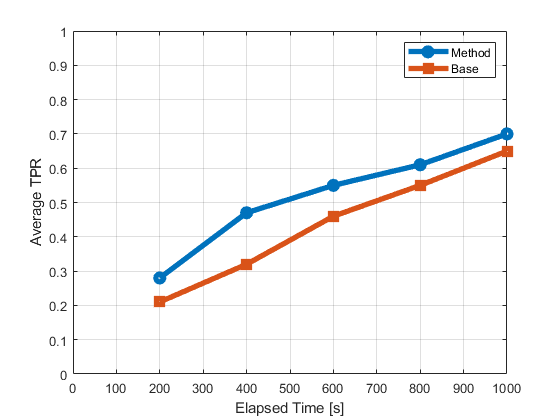
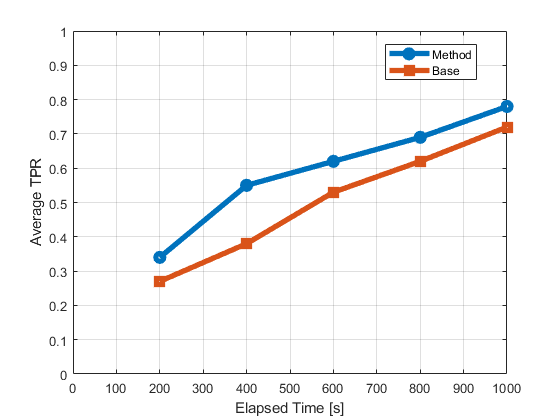
4-4 میانگین نرخ پردازش تراکنش‌ها

نرخ پردازش تراکنش‌ها یکی دیگر از شاخص‌های کلیدی عملکرد شبکه است که نشان می‌دهد چه میزان از درخواست‌ها یا جریان‌های ارسالی در واحد زمان به‌طور موفقیت‌آمیز پردازش و تکمیل می‌شوند. این معیار به‌ویژه برای شبکه‌هایی که میزبان سرویس‌های تعاملی یا حجم بالای تراکنش‌های هم‌زمان هستند اهمیت دارد، زیرا بیانگر کارایی سیستم در مدیریت بار و جلوگیری از ایجاد صف‌های طولانی در مسیر انتقال داده است.

نتایج حاصل از شبیه‌سازی در هر دو توپولوژی ۹ سوئیچی و ۱۳ سوئیچی حاکی از برتری محسوس روش پیشنهادی نسبت به روش پایه است. در توپولوژی ۹ سوئیچی، در ابتدای بازه زمانی (۲۰۰ ثانیه)، روش پیشنهادی TPR معادل 0.35را ثبت کرده است، در حالی‌که روش پایه مقدار پایین‌تری حدود 0.28 را نشان می‌دهد. با گذشت زمان و افزایش بار شبکه، این اختلاف عملکرد همچنان پابرجا می‌ماند. برای نمونه، در زمان ۶۰۰ ثانیه، TPR روش پیشنهادی به حدود 0.63 رسیده، در حالی‌که مقدار روش پایه حدود 0.55 است. در پایان آزمایش (۱۰۰۰ ثانیه)، روش پیشنهادی به حدود 0.80 ارتقا یافته و روش پایه مقدار پایین‌تری برابر 0.72 را ثبت کرده است. این نتایج نشان می‌دهد که الگوریتم پیشنهادی با در نظر گرفتن معیارهای چندگانه و انتخاب مسیرهای کم‌ترافیک‌تر، توانسته کارایی پردازش تراکنش‌ها را به طور پایدار بهبود دهد.

در توپولوژی ۱۳ سوئیچی نیز روند مشابهی مشاهده می‌شود، هرچند به دلیل افزایش پیچیدگی شبکه، مقادیر مطلق در مقایسه با سناریوی ۹ سوئیچی کمی پایین‌تر است. در زمان ۲۰۰ ثانیه، روش پیشنهادی مقدار 0.29 و روش پایه 0.21 را ثبت کرده‌اند. در میانه آزمایش (۶۰۰ ثانیه)، مقادیر به ترتیب به 0.55 و 0.47 افزایش یافته و در پایان بازه زمانی (۱۰۰۰ ثانیه)، روش پیشنهادی به حدود 0.70 و روش پایه به حدود 0.66 رسیده‌اند. این نتایج تأکید می‌کنند که حتی در شبکه‌های بزرگ‌تر و پرترافیک‌تر، روش پیشنهادی قادر است با تخصیص بهینه منابع و انتخاب هوشمندانه مسیرها، نرخ تکمیل تراکنش‌ها را افزایش دهد و پایداری عملکرد شبکه را تضمین نماید.

در مجموع، مقایسه دو توپولوژی نشان می‌دهد که روش پیشنهادی در تمامی بازه‌های زمانی و تحت هر دو سناریو عملکرد بهتری نسبت به روش پایه داشته است. این بهبود که در برخی موارد تا ۱۰ درصد افزایش TPR را به همراه دارد، مستقیماً ناشی از قابلیت تصمیم‌گیری چندمعیاره و بازپیکربندی پویا مسیرها در الگوریتم پیشنهادی است. چنین قابلیتی به‌ویژه در کاربردهایی مانند مراکز داده، سرویس‌های ابری و سامانه‌های استریم ویدئویی که نیازمند پردازش سریع و بی‌وقفه تراکنش‌ها هستند، اهمیت دوچندان دارد.



شکل 10- میانگین نرخ پردازش تراکنش ها در سناریو دوم

شکل 9- میانگین نرخ پردازش تراکنش ها در سناریو اول

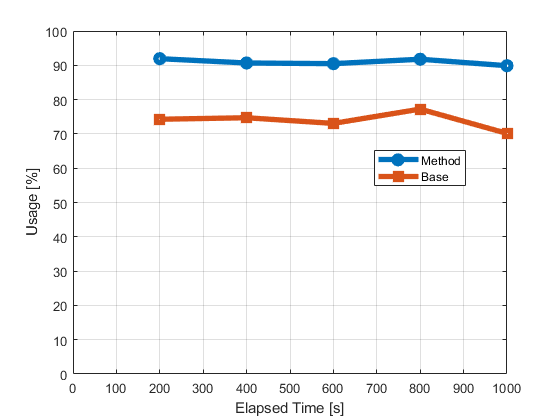
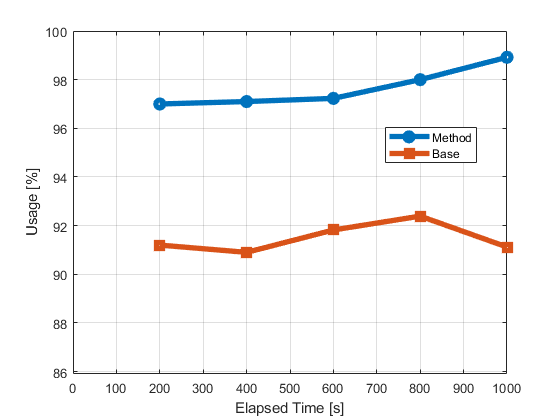
5-4 میانگین استفاده از منابع

استفاده از منابع (Usage) معیاری کلیدی برای نشان دادن میزان بهره‌برداری مؤثر از ظرفیت شبکه است. این معیار مشخص می‌کند چه درصدی از منابع موجود (مانند پهنای‌باند و توان پردازشی) در طول زمان به‌طور واقعی مورد استفاده قرار گرفته است. هرچه این مقدار بالاتر و پایدارتر باشد، نشان‌دهنده مدیریت بهینه جریان‌ها، توزیع هوشمند بار و کاهش هدررفت منابع شبکه است.

در توپولوژی ۹ سوئیچی (شکل اول)، روند تغییرات نشان می‌دهد که روش پیشنهادی در تمام بازه‌های زمانی عملکردی به‌مراتب بهتر از روش پایه داشته است. در آغاز آزمایش (۲۰۰ ثانیه)، مقدار استفاده از منابع در روش پیشنهادی حدود ۹۷٪ و در روش پایه نزدیک به ۹۱٪ ثبت شده است. با گذر زمان و افزایش بار، روش پیشنهادی توانسته استفاده از منابع را در سطح بسیار بالایی حفظ کند و حتی در انتهای آزمایش (۱۰۰۰ ثانیه) به حدود ۹۹٪ برسد، در حالی‌که روش پایه همچنان حول ۹۱٪ نوسان کرده و نتوانسته بهبود معناداری ایجاد کند. این اختلاف تقریباً ۸ درصدی نشان می‌دهد که رویکرد ما توانسته از ظرفیت موجود به نحو کامل‌تری بهره‌برداری کند و ترافیک را بدون هدررفت روی لینک‌ها توزیع نماید.

در توپولوژی ۱۳ سوئیچی (شکل دوم)، به دلیل پیچیدگی بیشتر و افزایش مسیرهای جایگزین، اختلاف میان دو روش باز هم مشهود است. روش پیشنهادی در آغاز آزمایش حدود ۹۲٪ از منابع را به‌کار گرفته، در حالی‌که روش پایه مقدار بسیار پایین‌تری در حدود ۷۴٪ ثبت کرده است. این شکاف عملکردی در طول آزمایش نیز پایدار مانده است. در پایان بازه (۱۰۰۰ ثانیه)، روش پیشنهادی همچنان حدود ۹۰٪ استفاده از منابع را نشان می‌دهد، اما روش پایه حتی با اندکی افت، به حدود ۷۱٪ سقوط کرده است. این نتایج نشان می‌دهند که در توپولوژی‌های بزرگ‌تر، الگوریتم پیشنهادی توانسته منابع را با کارایی نزدیک به حداکثر ظرفیت مدیریت کند، در حالی‌که روش پایه به دلیل تصمیم‌گیری‌های ساده و ایستا، بخش قابل‌توجهی از منابع شبکه را بلااستفاده باقی گذاشته است.

به‌طور کلی، مقایسه دو سناریو بیانگر آن است که الگوریتم پیشنهادی نه‌تنها توانسته استفاده بهینه از منابع را در شبکه کوچک‌تر تضمین کند، بلکه در شبکه بزرگ‌تر و پیچیده‌تر نیز کارایی بالای خود را حفظ کرده است. این موضوع به‌ویژه برای کاربردهایی همچون مراکز داده و شبکه‌های ابری حیاتی است، چراکه مدیریت ناکارآمد منابع در این محیط‌ها به‌سرعت منجر به هدررفت ظرفیت و افت کیفیت سرویس خواهد شد. در مقابل، الگوریتم ما با اتکا به انتخاب مسیر چندمعیاره و بازپیکربندی پویا، استفاده از منابع را در سطحی نزدیک به بهینه نگه داشته و بدین ترتیب اثربخشی کلی شبکه را ارتقا داده است.



شکل 11- میانگین استفاده از منابع در سناریو اول

شکل 12- میانگین استفاده از منابع در سناریو دوم

5- بحث و تحلیل

**1-5 نوآوری‌ها و دستاوردهای کلیدی**

یافته‌های این پژوهش نشان می‌دهد که الگوریتم مسیریابی پویا مبتنی بر هوش مصنوعی در محیط شبیه‌سازی NS2 توانسته است بهبود قابل توجهی نسبت به روش‌های سنتی مبتنی بر شمارش گام و حتی رویکرد مسیریابی آگاه از QoS کنترلر Floodlight ارائه دهد. این بهبودها به‌ویژه در شرایط پرترافیک بیشتر نمایان می‌شوند و اهمیت عملی این رویکرد را برای شبکه‌های واقعی برجسته می‌سازند.

یکی از مهم‌ترین نوآوری‌های این تحقیق، استفاده از مدل رگرسیون لجستیک به عنوان یک الگوریتم سبک‌وزن و تفسیرپذیر است که امکان طبقه‌بندی سطوح ترافیک شبکه را با دقت بالا فراهم می‌آورد. برخلاف مدل‌های پیچیده‌تر، رگرسیون لجستیک هزینه محاسباتی کمی دارد و به راحتی می‌تواند در کنترلرهای SDN ادغام شود.

**2-5 محدودیت‌ها**

اگرچه نتایج شبیه‌سازی در محیط NS2 امیدوارکننده است، باید توجه داشت که این محیط تمامی پیچیدگی‌ها و پویایی‌های شبکه‌های دنیای واقعی را بازتاب نمی‌دهد. نبود فاصله فیزیکی میان تجهیزات و اتکا به شرایط مصنوعی ترافیک ممکن است کاربرد مستقیم نتایج را محدود کند. همچنین مجموعه داده ایجادشده در NS2 به صورت مصنوعی تولید شده و ممکن است در مواجهه با داده‌های واقعی، نتایج متفاوتی حاصل شود.

این تحقیق عمدتاً بر توپولوژیNSFNET و ترافیک TCP متمرکز بوده است. در حالی که در شبکه‌های واقعی، انواع متنوعی از پروتکل‌ها و معماری‌ها وجود دارند که می‌توانند بر عملکرد الگوریتم اثر بگذارند. از سوی دیگر، فرض امنیت کامل در ارتباط میان کنترلر و سوئیچ‌ها در این مطالعه لحاظ شده است، در حالی که در دنیای واقعی تهدیدات امنیتی می‌توانند کارایی کنترلر را تحت‌تأثیر قرار دهند.

6- جمع بندی

در این مطالعه یک الگوریتم مسیریابی پویا مبتنی بر هوش مصنوعی در بستر شبکه‌های نرم‌افزارمحور و در محیط شبیه‌سازی NS2 ارائه و ارزیابی شد. هدف اصلی رویکرد پیشنهادی، ارتقای کیفیت خدمات شبکه از طریق انتخاب هوشمندانه مسیرها و مدیریت بهینه ترافیک بود. الگوریتم پیشنهادی با بهره‌گیری از مدل رگرسیون لجستیک و استفاده از مجموعه‌ای چندمعیاره شامل شاخص‌هایی مانند تأخیر، نرخ تحویل بسته، گذردهی، نرخ پردازش تراکنش‌ها و میزان استفاده از منابع، توانست مسیرهای کم‌ترافیک‌تر و کارآمدتر را شناسایی و انتخاب کند.

نتایج شبیه‌سازی نشان داد که این الگوریتم در مقایسه با روش‌های پایه مبتنی بر شمارش گام و رویکرد مسیریابی آگاه از QoS در کنترلر Floodlight، عملکرد بسیار بهتری دارد. به‌طور خاص، کاهش معنادار در تأخیر، افزایش قابل توجه در نرخ تحویل بسته، بهبود گذردهی، ارتقای نرخ پردازش تراکنش‌ها و استفاده کارآمدتر از منابع شبکه از مهم‌ترین دستاوردهای این مطالعه بودند.

با وجود این دستاوردها، باید اذعان داشت که نتایج در محیط شبیه‌سازی NS2 به‌دست آمده و ممکن است بازتاب کامل پیچیدگی‌های شبکه‌های واقعی نباشد. با این حال، یافته‌های تحقیق حاضر چشم‌اندازی روشن برای توسعه و پیاده‌سازی الگوریتم‌های مسیریابی مبتنی بر هوش مصنوعی در شبکه‌های واقعی فراهم می‌کند. این الگوریتم‌ها می‌توانند نقش مهمی در بهینه‌سازی منابع، تضمین کیفیت خدمات و پشتیبانی از کاربردهای نوظهور با نیازهای بالای QoS ایفا کنند. در نهایت، این مطالعه نشان می‌دهد که ادغام هوش مصنوعی و SDN نه تنها امکان‌پذیر است، بلکه می‌تواند گام مؤثری در جهت ساخت نسل آینده شبکه‌های هوشمند، کارآمد و کاربرمحور باشد.

همچنین برای تحقیقات آینده، چند مسیر مهم پیشنهاد می‌شود:

* بررسی عملکرد الگوریتم در توپولوژی‌های بزرگ‌تر و متنوع‌تر
* ارزیابی تحت انواع مختلف ترافیک UDP، ICMP و ترکیبی
* بهره‌گیری از مدل‌های یادگیری عمیق برای پیش‌بینی دقیق‌تر ترافیک
* تحلیل مصرف منابع کنترلر (CPU) و حافظه
* و همچنین آزمون الگوریتم در محیط‌های عملیاتی واقعی به منظور ارزیابی میزان انطباق و پایداری آن

**مراجع**

1. Software-Defined Network [↑](#footnote-ref-1)
2. Delay [↑](#footnote-ref-2)
3. Packet Delivery Ratio [↑](#footnote-ref-3)
4. Throughput [↑](#footnote-ref-4)
5. Transaction Processing Rate [↑](#footnote-ref-5)
6. Usage of Resources [↑](#footnote-ref-6)
7. Hop Count [↑](#footnote-ref-7)
8. Artificial Intelligence [↑](#footnote-ref-8)
9. Machine Learning [↑](#footnote-ref-9)
10. Quality of Service [↑](#footnote-ref-10)
11. QoS-aware [↑](#footnote-ref-11)
12. Ant Colony Optimization [↑](#footnote-ref-12)
13. Particle Swarm Optimization [↑](#footnote-ref-13)
14. Quality of Experience [↑](#footnote-ref-14)
15. Content Delivery Network [↑](#footnote-ref-15)
16. AI based routing in software-defined networks [↑](#footnote-ref-16)
17. Southbound [↑](#footnote-ref-17)
18. Match–Action [↑](#footnote-ref-18)
19. Northbound [↑](#footnote-ref-19)
20. Flow Table [↑](#footnote-ref-20)
21. multithreaded [↑](#footnote-ref-21)
22. k-shortest paths [↑](#footnote-ref-22)
23. Route Flapping [↑](#footnote-ref-23)