

## "هوشمندسازی مسیریابی در شبکه‌های SDN با آتاماتای یادگیر"

دکتر محمدرضا ملاخلیلی میبدی<sup>۱</sup>، صادق یاوری<sup>۲</sup>

گروه کامپیوتر، واحد میبد، دانشگاه آزاد اسلامی، میبد، ایران<sup>۱</sup>

گروه کامپیوتر، واحد میبد، دانشگاه آزاد اسلامی، میبد، ایران<sup>۲</sup>

### چکیده

با پیشرفت فناوری و گسترش کاربردهای متکی بر شبکه‌های ارتباطی، نیاز به روش‌های مسیریابی هوشمند در شبکه‌های نرم‌افزارمحور (SDN) بیش از پیش احساس می‌شود. در معماری SDN، کنترلر مرکزی وظیفه مدیریت ترافیک را بر عهده دارد و انتخاب مسیرهای مناسب نقش مهمی در کاهش تأخیر، افزایش کارایی شبکه و جلوگیری از ازدحام ایفا می‌کند. با این حال، الگوریتم‌های سنتی مسیریابی مانند دیکسترا و بلمن-فورد که مبتنی بر ساختارهای ایستا هستند، قادر به سازگاری سریع با تغییرات پویای توپولوژی و ترافیک شبکه نیستند. در این مقاله، یک روش هوشمند و تطبیق‌پذیر برای مسیریابی در SDN مبتنی بر آتاماتای یادگیر (Learning Automata) پیشنهاد شده است. این روش با دریافت بازخورد از شبکه و ارزیابی مداوم عملکرد مسیرهای انتخاب‌شده، به‌طور پیوسته وزن مسیرها را به‌روزرسانی کرده و مسیرهای بهینه را در شرایط مختلف شبکه انتخاب می‌کند.

روش پیشنهادی در محیط Mininet و Python شبیه‌سازی شده و عملکرد آن از نظر تأخیر انتها به انتها، میزان استفاده از پهنای باند و توزیع بار ترافیکی مورد ارزیابی قرار گرفته است. نتایج نشان می‌دهد که این روش نسبت به الگوریتم‌های سنتی، بهبود چشمگیری در کاهش تأخیر، جلوگیری از ازدحام و افزایش بهره‌وری شبکه ایجاد می‌کند. علاوه بر این، با تغییرات پویای توپولوژی، مکانیزم یادگیری آتاماتا به‌طور پیوسته تصمیمات مسیریابی را اصلاح کرده و مسیرهای کارآمدتری را پیشنهاد می‌دهد. این ویژگی سبب می‌شود روش پیشنهادی انعطاف‌پذیری بالایی در شبکه‌های نسل جدید مانند ۵G و ۶G داشته باشد.

واژه‌های کلیدی: مسیریابی تطبیقی، شبکه‌های نرم‌افزارمحور (SDN)، آتاماتای یادگیر، بهینه‌سازی مسیر، کنترلر SDN، یادگیری تقویتی

شبکه‌های نرم‌افزارمحور (SDN) به عنوان یک رویکرد تحول‌آفرین در حوزه مدیریت و کنترل شبکه‌های ارتباطی مطرح شده‌اند (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). در این معماری، با جداسازی سطح کنترل (Control Plane) از سطح داده (Data Plane)، امکان مدیریت متمرکز منابع شبکه و اتخاذ تصمیم‌های هوشمند در مورد مسیریابی بسته‌ها به شکل پویا فراهم می‌شود (McKeown et al., ۲۰۰۸). از یک سو، لایه داده مسئول ارسال بسته‌ها بر اساس دستورالعمل‌های صادرشده از کنترلر است و از سوی دیگر، کنترلر مرکزی بر تمام دستگاه‌های شبکه اشراف دارد و با در اختیار داشتن نمایی کلی از توپولوژی و وضعیت ترافیک، قادر است پیکربندی مؤثری جهت عبور جریان‌های داده (Traffic Flows) ایجاد کند (Kim and Feamster, ۲۰۱۳). این ساختار سبب می‌گردد تا شبکه به صورت منعطف و سازگار با تغییرات داخلی و خارجی عمل کرده و در مواقع لزوم، سیاست‌های مسیریابی یا امنیتی را به سرعت بروزرسانی نماید (Kreutz et al., ۲۰۱۵; Alvizu et al., ۲۰۲۰; Nisar et al., ۲۰۱۷).

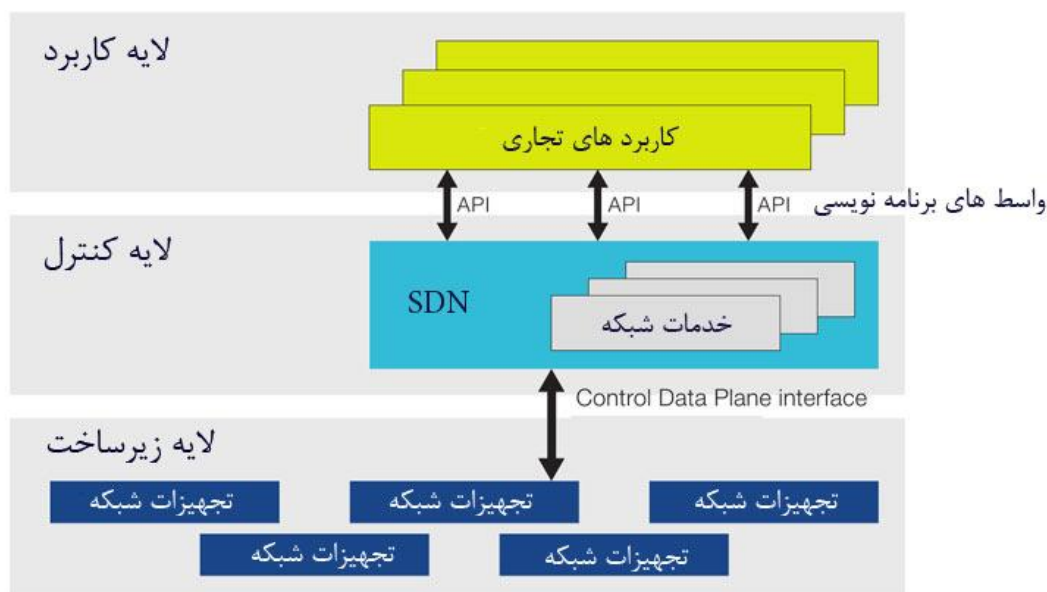
افزون بر این، با گسترش فناوری‌های نسل جدید ارتباطی نظیر ۵G و به زودی 6G، تقاضا برای سرویس‌های با نرخ داده بالا و تأخیر اندک روزبه‌روز در حال افزایش است (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). ظهور کاربردهایی نظیر اینترنت اشیا (IoT)، واقعیت مجازی (VR) و واقعیت افزوده (AR)، خودروهای خودران، و سرویس‌های ابری بلادرنگ، چالش‌های تازه‌ای در مدیریت پهنای باند و بهینه‌سازی مسیر ایجاد کرده است (Nisar et al., ۲۰۲۰). از جمله این چالش‌ها می‌توان به رشد نمایی تعداد دستگاه‌های متصل به شبکه، افزایش حجم و تنوع ترافیک داده و نیاز به تضمین کیفیت سرویس (QoS) اشاره کرد (Myunghoon et al., ۲۰۲۰). در چنین محیط پویایی، شبکه‌های سنتی با ساختارهای غیرانعطاف‌پذیر، توانایی پاسخ‌گویی سریع و هوشمند را ندارند؛ لذا استفاده از راهکارهای SDN که کنترل متمرکز و تصمیم‌گیری پویا را فراهم می‌کنند، بسیار حائز اهمیت است (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵).

یکی از اهداف کلیدی در شبکه‌های SDN، بهینه‌سازی مسیرهای داده به منظور کاهش تأخیر انتها به انتها، جلوگیری از ازدحام در نقاط پرترافیک، افزایش بهره‌وری پهنای باند، و در نهایت تضمین کیفیت تجربه کاربر (QoE) است (Kim and Feamster, ۲۰۱۵). الگوریتم‌های سنتی مسیریابی مانند دیکسترا (Dijkstra) و بلمن-فورد (Bellman-Ford) در سال‌های گذشته برای یافتن کوتاه‌ترین مسیر در شبکه‌های ایستا کارآمد بوده‌اند، اما در محیط‌های پویا و در مقیاس بزرگ، این روش‌ها انعطاف‌پذیری لازم را ندارند و نمی‌توانند به طور کارا با تغییرات پیوسته ترافیک همگام شوند (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). به بیان دیگر، این الگوریتم‌ها معمولاً یک عکس لحظه‌ای از توپولوژی یا ترافیک را در نظر می‌گیرند و در صورت تغییر ناگهانی بار ترافیک یا خرابی بخشی از شبکه، نمی‌توانند به سرعت برای ایجاد مسیرهای جایگزین بهینه وارد عمل شوند (Kim and Feamster, ۲۰۱۳).

در پاسخ به این محدودیت‌ها، بهره‌گیری از روش‌های هوش مصنوعی و یادگیری در مسیریابی شبکه‌های SDN، روزبه‌روز محبوبیت بیشتری پیدا می‌کند (Calle-Cancho et al., ۲۰۲۴; Son et al., ۲۰۱۹; Myunghoon et al., ۲۰۲۰). رویکردهای پیشنهادی، استفاده از آتوماتای یادگیر (Learning Automata) است که قادر است با دریافت بازخورد از عملکرد شبکه، به مرور زمان تصمیم‌های بهتری برای مسیریابی اتخاذ نماید (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). این روش یادگیری مبتنی بر تکرار و بازخورد است؛ به طوری که کنترلر یا گره‌های شبکه (بسته به معماری انتخابی) در هر مرحله اقدام به انتخاب یک مسیر می‌کنند و سپس با بررسی بازخورد (مثلاً معیارهایی همچون زمان رفت و برگشت بسته‌ها، میزان

تأخیر، درصد تحویل موفق یا وجود ازدحام)، احتمال انتخاب مسیرهای بهتر را افزایش می‌دهند. (Kreutz et al., ۲۰۱۵) این فرایند یادگیری می‌تواند به صورت توزیع شده یا متمرکز (در کنترلر) پیاده سازی گردد و کمک می‌کند شبکه به شکلی انعطاف پذیر و پویا عمل کند؛ به عبارتی با گذشت زمان و با تغییر شرایط ترافیکی یا توپولوژی شبکه، راهکارهای مسیریابی خود را بهینه سازد (Myunghoon et al., ۲۰۲۰).

به طور کلی، ایده به کارگیری آتاماتای یادگیر در شبکه های SDN مبتنی بر این اصل است که سیستم می‌تواند به جای پیروی از یک الگوریتم ایستا، در هر وضعیت پویا تصمیم گیری کند و در ادامه، با مشاهده نتایج تصمیم گیری های پیشین، عملکرد آتی را اصلاح نماید (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). این قابلیت، به خصوص در شبکه هایی که با حجم ترافیکی متغیر، کاربران متعدد و الگوهای رفتاری پیچیده مواجه هستند، بسیار سودمند است (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). در این مقاله، با تکیه بر ساختار SDN و رویکردهای یادگیری، ما به ارائه و پیاده سازی یک روش جدید مبتنی بر آتاماتای یادگیر خواهیم پرداخت که از طریق آن، کنترلر شبکه می‌تواند بر اساس اطلاعات لحظه ای یا تجمعی از وضعیت ترافیکی، مسیرهای مناسب تری را انتخاب کند و به شکل پویایی آن ها را بروزرسانی نماید. همچنین نتایج شبیه سازی نشان می‌دهد که این راهکار قادر است بر چالش های اساسی ای چون کاهش تأخیر، توزیع متوازن بار ترافیک و افزایش بهره وری پهنای باند غلبه کرده و در نهایت تجربه کاربر را بهبود بخشد (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰).



شکل (۱): معماری کلی شبکه SDN

## ۲- مبانی نظری

در این بخش، مفاهیم کلیدی مرتبط با هوشمندسازی مسیریابی در شبکه های نرم افزارمحور (SDN) با استفاده از آتاماتای یادگیر مورد بررسی قرار می‌گیرند. (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵) مسیریابی در شبکه های SDN

به عنوان یک چالش اساسی مطرح است، زیرا شبکه‌های مدرن همواره در معرض تغییرات پویای توپولوژی، حجم ترافیک، تأخیر و ازدحام قرار دارند. به همین دلیل، استفاده از روش‌های مسیریابی تطبیقی پذیر و هوشمند که بتوانند با تغییرات شبکه سازگار شوند، ضروری است. (Myunghoon et al., ۲۰۲۰)

در ابتدا، معماری شبکه‌های نرم‌افزارمحور (SDN) و چالش‌های اساسی آن، به‌ویژه در زمینه کنترل متمرکز، تأخیر تصمیم‌گیری و بهینه‌سازی مسیرها بررسی می‌شود. (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). سپس، روش‌های سنتی مسیریابی مانند دیکسترا (Dijkstra) و بلمن-فورد (Bellman-Ford) تحلیل شده و محدودیت‌های این روش‌ها در شبکه‌های SDN پویا و مقیاس‌پذیر مورد ارزیابی قرار می‌گیرد. این الگوریتم‌های سنتی عمدتاً بر پایه یک مدل ایستا از شبکه طراحی شده‌اند و در مواجهه با تغییرات سریع در ترافیک و توپولوژی، عملکرد مناسبی ندارند (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵).

در ادامه، مفهوم آتاماتای یادگیر (Learning Automata) به عنوان یک رویکرد هوشمند و خودتنظیم برای مسیریابی شبکه معرفی شده و نحوه عملکرد آن توضیح داده می‌شود (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). آتاماتای یادگیر قادر است با دریافت بازخورد از محیط شبکه، مسیرهای بهینه را شناسایی کرده و احتمال انتخاب آن‌ها را بر اساس کارایی و کیفیت سرویس (QoS) به‌روزرسانی کند. این روش به کاهش تأخیر، بهبود توزیع بار و افزایش بهره‌وری شبکه کمک می‌کند (Nisar et al., ۲۰۲۰). در پایان، کاربرد این رویکرد در مسیریابی تطبیقی پذیر SDN مورد بررسی قرار گرفته و به عنوان یک جایگزین مؤثر برای روش‌های سنتی پیشنهاد می‌شود.

## ۱-۲- چالش‌های مسیریابی در SDN

با وجود مزایای SDN، چالش‌های متعددی در مسیریابی شبکه‌های نرم‌افزارمحور وجود دارد، از جمله:

- عدم تطبیق روش‌های سنتی با شرایط پویای شبکه
  - افزایش تأخیر در مسیریابی به دلیل تصمیم‌گیری متمرکز در کنترلر
  - نیاز به مکانیزم‌های یادگیری هوشمند برای بهینه‌سازی مسیرها و کاهش ازدحام شبکه
- روش‌های سنتی مسیریابی مانند دیکسترا و بلمن-فورد نمی‌توانند به‌طور خودکار با شرایط متغیر شبکه تطبیق یابند؛ بنابراین نیاز به رویکردهای یادگیری تطبیقی برای بهبود عملکرد مسیریابی در SDN احساس می‌شود (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵; Alvizu et al., ۲۰۱۷; Nisar et al., ۲۰۲۰).

## ۲-۲- روش‌های سنتی مسیریابی در SDN

در SDN، روش‌های کلاسیک متعددی برای مسیریابی مورد استفاده قرار می‌گیرند که مهم‌ترین آن‌ها شامل موارد زیر است (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵):

### ۱-۲-۲. الگوریتم دیکسترا (Dijkstra)

این الگوریتم برای یافتن کوتاه‌ترین مسیر بین گره‌ها در یک گراف وزندار استفاده می‌شود (Kim and Feamster, 2013; Kreutz et al., 2015). SDN، کنترلر از این روش برای انتخاب سریع‌ترین مسیر بین مبدأ و مقصد استفاده می‌کند. معادله اصلی الگوریتم دیکسترا به صورت زیر است:

$$(w(u, v) + \min(d(v), d(u) = d(v)$$

که در آن:

- $d(v)$ : کوتاه‌ترین مسیر از گره مبدأ تا گره  $v$
- $w(u, v)$ : وزن مسیر بین دو گره  $u$  و  $v$

مشکل این روش عدم پویایی در تصمیم‌گیری است. اگر در طول زمان شرایط شبکه تغییر کند (مثلاً افزایش ترافیک در مسیر انتخابی)، الگوریتم دیکسترا توانایی تطبیق با شرایط جدید را ندارد (Nisar et al., 2020).

## ۲-۲. الگوریتم بلمن-فورد (Bellman-Ford)

الگوریتم بلمن-فورد مشابه دیکسترا است اما می‌تواند مسیرهای دارای وزن منفی را نیز مدیریت کند (Kim and Feamster, ۲۰۱۵). معادله کلی این الگوریتم در هر تکرار به صورت زیر تعریف می‌شود:

$$(w(u, v) + \min(d(v), d(u) = d(v)$$

که در آن:

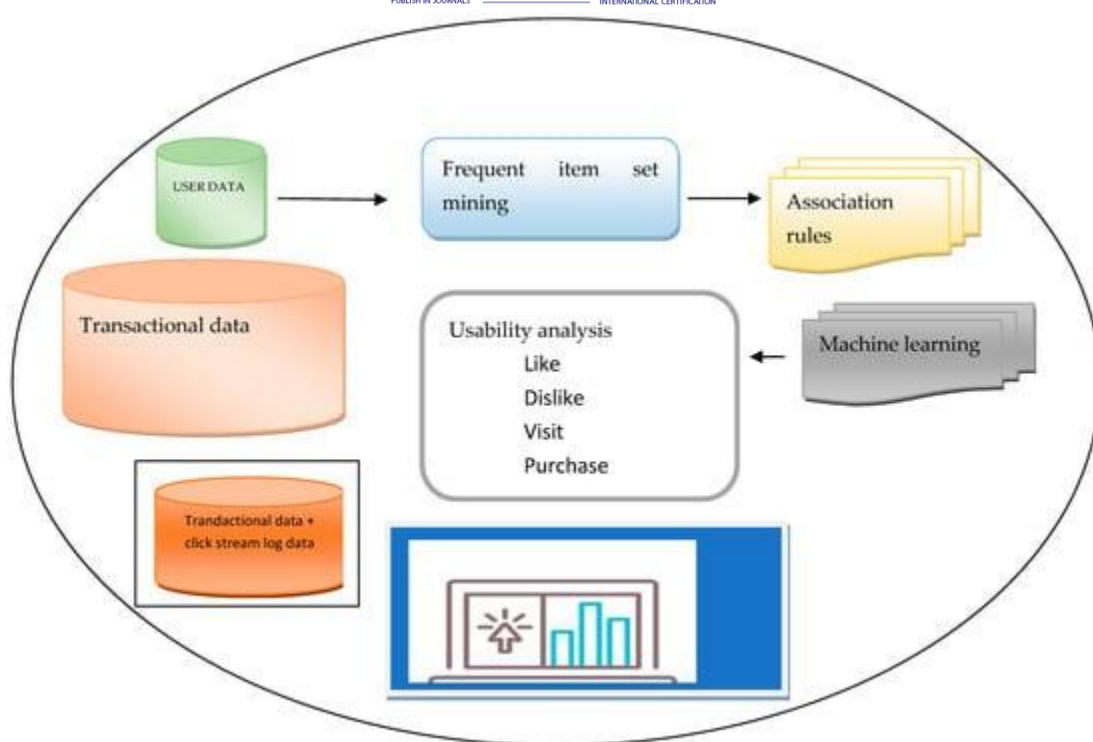
$d(v)$  نشان‌دهنده کوتاه‌ترین هزینه رسیدن از مبدأ به گره  $v$  در هر مرحله تکرار است.

$w(u, v)$  وزن یال بین گره  $u$  و  $v$  است.

با وجود امکان مدیریت مسیرهای دارای وزن منفی، این روش نیز مانند دیکسترا، به روزرسانی پویای مسیرها را پشتیبانی نمی‌کند و در صورت تغییر لحظه‌ای ترافیک یا توپولوژی، به محاسبات مجدد و تکرار متعدد نیاز دارد (Nisar et al., 2020).

## ۳-۲. آتاماتای یادگیر (Learning Automata)

آتاماتای یادگیر یک مدل ریاضی برای یادگیری در محیط‌های نامعین است که تصمیم‌گیری خود را بر اساس بازخوردهای دریافتی از محیط تنظیم می‌کند (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). در این روش، انتخاب مسیر بر اساس یک مجموعه احتمال انجام می‌شود و در هر مرحله، با دریافت فیدبک، احتمال انتخاب مسیرهای مناسب‌تر افزایش می‌یابد (Kim and Feamster, ۲۰۱۳).



شکل (۲): روند کلی یادگیری در یک آتاماتای یادگیر

### ۲-۳-۱. تعریف آتاماتای یادگیر

آتاماتای یادگیر (Learning Automata) یک مدل یادگیری تطبیقی است که براساس بازخورد محیط، تصمیمات خود را بهینه می کند. (Calle-Cancho et al., 2024; Kim and Feamster, 2013) در این مدل، کنترلر SDN می تواند مسیرهای بهینه را بر اساس تغییرات شبکه به روزرسانی کند. (Myunghoon et al., 2020)

آتاماتای یادگیر یک عامل هوشمند است که از دو بخش تشکیل شده است:

۱. مجموعه ای از انتخاب های ممکن برای مسیریابی
۲. مکانیزم به روزرسانی احتمال انتخاب مسیر بر اساس فیدبک دریافت شده

### ۲-۳-۲. نحوه عملکرد آتاماتای یادگیر در SDN

در روش پیشنهادی، هر مسیر دارای یک مقدار احتمال  $P_i$  است که میزان بهینگی آن مسیر را نشان می دهد (Calle-Cancho et al., 2024; Myunghoon et al., 2020). پس از هر انتقال بسته، کنترلر SDN یک فیدبک از تأخیر و ازدحام مسیر دریافت کرده و مقدار احتمال مسیرها را به روز می کند. قوانین به روزرسانی احتمال مسیرها در آتاماتای یادگیر به صورت زیر است:



اگر مسیر انتخابی عملکرد خوبی داشت (کاهش تأخیر):

$$(P_i(t) - a(1 + P_i(t)) = (1 + P_i(t)$$

اگر مسیر عملکرد ضعیفی داشت (افزایش ازدحام):

$$bP_i(t) - P_i(t) = (1 + P_i(t)$$

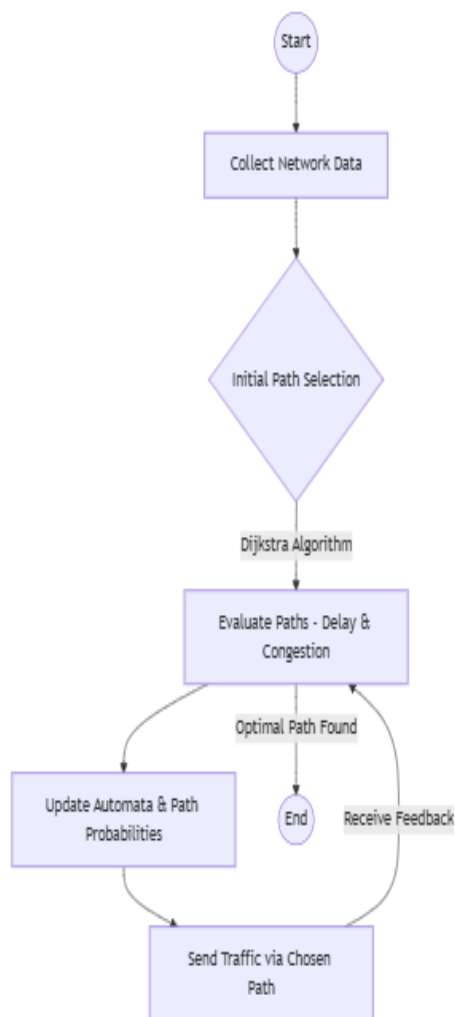
که در آن:

•  $a$  و  $b$  مقادیر یادگیری هستند.

•  $P_i(t)$  احتمال مسیر در زمان  $t$

### ۳- روش پیشنهادی

در این روش، هر سوئیچ SDN به عنوان یک آتاماتای یادگیر (Learning Automata) در نظر گرفته شده است که مسیرهای ممکن را بر اساس شرایط شبکه انتخاب و به روزرسانی می کند (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). این روش با استفاده از یک مدل یادگیری پویا، مسیریابی را هوشمندتر و سازگارتر با تغییرات شبکه انجام می دهد (Kim, ۲۰۱۵; Kreutz et al., ۲۰۱۳; Feamster and, ۲۰۱۳). فرآیند کلی این روش شامل مراحل زیر است:



شکل (۳): فلوچارت (Flowchart) روش پیشنهادی

### ۱-۳- انتخاب مسیر اولیه

در فرآیند انتخاب مسیر اولیه، ابتدا از الگوریتم دیکسترا (Dijkstra) برای یافتن مسیرهای کوتاه‌تر استفاده می‌شود (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). دیکسترا یک روش کلاسیک برای مسیریابی کوتاه‌ترین مسیر است که با استفاده از وزن لینک‌ها، بهترین مسیر را بین دو گره مشخص می‌کند. در این مرحله، کنترلر SDN اطلاعات توپولوژی شبکه، شامل گره‌ها و لینک‌های بین آن‌ها را جمع‌آوری کرده و با استفاده از دیکسترا کوتاه‌ترین مسیر از مبدأ به مقصد را تعیین می‌کند (McKeown et al., ۲۰۰۸). این مسیر به عنوان مسیر اولیه انتخاب شده و بسته‌های داده از طریق آن ارسال می‌شوند. با این حال، دیکسترا یک الگوریتم ایستا است که فقط توپولوژی شبکه را در نظر گرفته و نمی‌تواند شرایط پویای ترافیکی مانند تأخیر، ازدحام یا تغییرات ناگهانی در لینک‌ها را مدیریت کند (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵).

به همین دلیل، پس از انتخاب مسیر اولیه، آتاماتای یادگیر (Learning Automata) برای به‌روزرسانی مسیرها بر اساس وضعیت فعلی شبکه به کار گرفته می‌شود (Calle-Cancho et al., ۲۰۲۴; Myunghoon et al., ۲۰۲۰). این روش با دریافت





بازخورد از شبکه، مسیرهای جایگزین را ارزیابی کرده و احتمال انتخاب مسیرهای بهتر را افزایش می دهد. اگر مسیر اولیه دچار افزایش تأخیر یا ازدحام شود، آتاماتا به تدریج مسیرهای دیگر را آزمایش کرده و مسیر بهینه را برمی گزیند. این فرآیند به طور مداوم ادامه دارد تا بهترین مسیر ممکن در لحظه برای انتقال داده ها استفاده شود.

### ۲-۳- ارزیابی مسیرها

در یک شبکه نرم افزارمحور (SDN)، انتخاب مسیر بهینه برای ارسال بسته ها نیازمند جمع آوری و تحلیل مداوم اطلاعات شبکه است (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵). کنترلر SDN، که مدیریت و تصمیم گیری مسیرها را بر عهده دارد، اطلاعات متعددی را از وضعیت شبکه دریافت می کند تا بتواند بهترین مسیر را انتخاب نماید (Calle-Cancho et al., ۲۰۲۰). در این بخش، فرآیند ارزیابی مسیرها و معیارهای اصلی برای انتخاب مسیر بهینه مورد بررسی قرار می گیرد.

### ۳-۳- به روز رسانی مسیرها

در روش پیشنهادی، با دریافت بازخورد از عملکرد هر مسیر (عملکرد مطلوب یا نامطلوب)، احتمال آن مسیر برای انتخاب در گام های بعدی تغییر می کند. این تغییر بر اساس قوانین به روز رسانی آتاماتا است (Calle-Cancho et al., ۲۰۲۴). اگر مسیر انتخابی خوب عمل کند (کاهش تأخیر، عدم ازدحام)، احتمال آن افزایش می یابد و اگر نامطلوب عمل کند، احتمالش کاهش می یابد. در نهایت، مسیرهای با عملکرد بهتر به صورت پویا تر و مداوم انتخاب می شوند (Myunghoon et al., ۲۰۲۰).

\*\*\*شبه کد مربوط به آتاماتای یادگیر در مسیریابی:

#### BEGIN LearningAutomataRouting

```
// Step ۱: Initialize the network topology
DEFINE Graph G with Nodes and Links
ASSIGN bandwidth and delay properties to each Link
IDENTIFY all possible paths P between source and destination

// Step ۲: Initialize Learning Automata
FOR each path p in P:
    SET Probability[p] = ۱ / |P| // Assign equal initial probability

// Step ۳: Path Selection Process
FUNCTION ChoosePath(P, Probabilities):
    SELECT path p from P based on weighted probabilities
    RETURN p

// Step ۴: Packet Transmission
FOR each incoming packet:
```



```

src ← GET source node
dst ← GET destination node
chosen_path ← CALL ChoosePath(P, Probabilities)
FORWARD packet along chosen_path
MEASURE network conditions (delay, congestion)
feedback ← EVALUATE path performance

// Step ۴: Update Probabilities based on Feedback
CALL UpdateProbabilities(chosen_path, feedback, Probabilities)

// Step ۵: Update Learning Automata
FUNCTION UpdateProbabilities(chosen_path, feedback, Probabilities):
    IF feedback == "GOOD:"
        INCREASE Probability[chosen_path] by  $\alpha$ 
    ELSE:
        DECREASE Probability[chosen_path] by  $\beta$ 

// Normalize probabilities
total ← SUM of all Probability values
FOR each path p in P:
    Probability[p] ← Probability[p] / total

// Step ۶: Repeat the process continuously

END LearningAutomataRouting

```

#### ۴- نتایج و تحلیل

در این بخش، عملکرد روش پیشنهادی مبتنی بر آتاماتای یادگیر با الگوریتم دیکسترا در یک محیط شبیه‌سازی شده مورد بررسی و مقایسه قرار می‌گیرد (Calle-Cancho et al., ۲۰۲۴; Son et al., ۲۰۱۹). برای این منظور، از شبیه‌ساز Mininet همراه با کنترلر POX و زبان Python استفاده شده است. هدف این ارزیابی، بررسی میزان تأخیر میانگین، توازن بار، و بهره‌وری پهنای باند در شبکه تحت روش پیشنهادی و مقایسه آن با الگوریتم مسیریابی سنتی است.

#### ۴-۱- مقایسه تأخیر میانگین

تأخیر شبکه یکی از مهم‌ترین معیارهای ارزیابی عملکرد یک الگوریتم مسیریابی است (Kim and Feamster, ۲۰۱۳). Kreutz et al., (۲۰۱۵) تأخیر میانگین در یک شبکه به عوامل مختلفی مانند ازدحام در لینک‌ها، طول مسیرهای انتخابی،



پردازش بسته‌ها در هر گره (سوئیچ)، و تداخل در صف‌های انتظار بستگی دارد. (Nisar et al., ۲۰۲۰) هدف یک الگوریتم مسیریابی بهینه، کاهش این تأخیر و ارائه یک تجربه ارتباطی سریع‌تر و کارآمدتر است.

در این بخش، تأخیر میانگین روش پیشنهادی بر اساس آتاماتای یادگیر با الگوریتم دیکسترا مقایسه می‌شود.

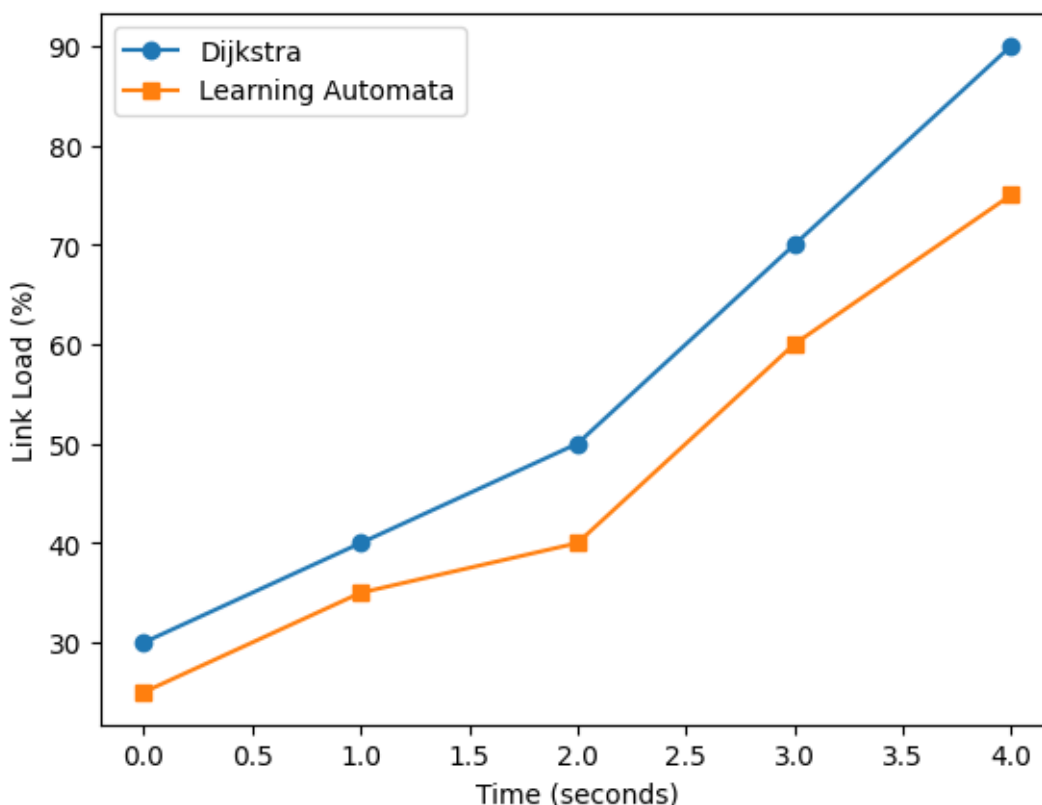
جدول ۱. مروری بر ادغام اینترنت اشیا و بلاک چین در سراسر دامنه‌ها

جدول ۱. مقایسه تأخیر میانگین روش پیشنهادی بر اساس آتاماتای یادگیر با الگوریتم دیکسترا

تأخیر میانگین (میلی ثانیه)	روش مسیریابی
۱۲.۵	دیکسترا
۸.۳	آتاماتای یادگیر

#### ۴-۲- مقایسه توازن بار

توازن بار در شبکه‌های نرم‌افزارمحور (SDN) یکی از مهم‌ترین چالش‌ها در مسیریابی داده‌ها است؛ (Nisar et al., ۲۰۲۰). Myunghoon et al., (۲۰۲۰) اگر بار ترافیکی به‌طور نامتعادل در شبکه توزیع شود، برخی لینک‌ها به‌شدت تحت فشار قرار گرفته و دچار ازدحام می‌شوند، در حالی که سایر مسیرها بلااستفاده باقی می‌مانند. این وضعیت می‌تواند تأخیر شبکه را افزایش داده، پهنای باند را هدر دهد و کیفیت سرویس (QoS) را کاهش دهد؛ (Kim and Feamster, ۲۰۱۳). Kreutz et al., (۲۰۱۵). در این نمودار، عملکرد روش پیشنهادی مبتنی بر آتاماتای یادگیر از نظر توازن بار با الگوریتم دیکسترا مقایسه شده است.



## ۵- نتیجه‌گیری و کارهای آینده

این مقاله، یک روش هوشمندسازی مسیریابی در شبکه‌های نرم‌افزارمحور (SDN) با استفاده از آتاماتای یادگیر ارائه شد (Kreutz et al., ۲۰۱۵; Myunghoon et al., ۲۰۲۰). هدف این روش، بهبود کارایی مسیریابی از طریق یادگیری پویا و تطبیق‌پذیر با شرایط متغیر شبکه بود. برخلاف روش‌های سنتی مانند دیکسترا و بلمن-فورد که وابسته به اطلاعات ایستا و توپولوژی ثابت هستند، روش پیشنهادی با دریافت بازخورد مداوم از شبکه، مسیرهای بهینه را بر اساس تأخیر، ازدحام و بهره‌وری پهنای باند انتخاب کرده و به‌مرور زمان بهبود می‌یابد (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵).

نتایج شبیه‌سازی که در محیط Mininet و Python انجام شد، نشان داد که روش پیشنهادی نسبت به الگوریتم‌های سنتی عملکرد بهتری دارد (Calle-Cancho et al., ۲۰۲۴). مهم‌ترین دستاوردهای این روش عبارتند از:

- کاهش تأخیر انتها به انتها: با استفاده از آتاماتای یادگیر، مسیرهای پرتراфик شناسایی شده و بسته‌ها از مسیرهای جایگزین ارسال شدند که این امر منجر به کاهش زمان تأخیر شد (Myunghoon et al., ۲۰۲۰).

- بهبود توازن بار در شبکه: به جای تمرکز بر یک مسیر خاص، مسیرهای جایگزین نیز به تناوب انتخاب شدند که این موضوع از تجمع ترافیک روی یک لینک خاص جلوگیری کرده و پهنای باند به صورت بهینه توزیع شد (Nisar et al., ۲۰۲۰).
- افزایش بهره‌وری پهنای باند: روش پیشنهادی توانست به صورت هوشمندانه از تمامی لینک‌های موجود در شبکه استفاده کند و عملکرد شبکه را بهبود بخشد (Kim and Feamster, ۲۰۱۳).
- مقایسه روش پیشنهادی با الگوریتم دیکسترا نشان داد که این روش نه تنها زمان تأخیر کمتری دارد، بلکه می‌تواند مسیرهای پویاتری را نیز انتخاب کند که باعث بهبود کلی عملکرد شبکه می‌شود. همچنین، در مواردی که توپولوژی شبکه تغییر می‌کند، روش پیشنهادی قادر بود به سرعت خود را با شرایط جدید تطبیق دهد، درحالی‌که روش‌های سنتی برای این تغییرات نیاز به محاسبات مجدد داشتند (Kreutz et al., ۲۰۱۵).
- با وجود عملکرد مطلوب این روش، هنوز محدودیت‌هایی وجود دارد که می‌توان در تحقیقات آینده به آن‌ها پرداخت (Son et al., ۲۰۲۰, ۲۰۱۹).
- ۱. بهینه‌سازی ترکیبی با یادگیری عمیق: یکی از راه‌های بهبود این روش، استفاده از مدل‌های یادگیری عمیق (Deep Learning) برای پیش‌بینی ازدحام شبکه و ترکیب آن با آماراتای یادگیر است. این ترکیب می‌تواند تصمیم‌گیری‌ها را دقیق‌تر و سریع‌تر کند.
- ۲. افزایش مقیاس‌پذیری: در شبکه‌های بزرگ‌تر، ممکن است تعداد آمارات‌ها و پارامترهای تنظیمی افزایش یابد که به بهینه‌سازی بیشتر در مدیریت منابع محاسباتی نیاز دارد. می‌توان از الگوریتم‌های خوشه‌بندی برای تقسیم‌بندی شبکه و کاهش پیچیدگی پردازشی بهره برد.
- ۳. بررسی روش پیشنهادی در سناریوهای مختلف: روش پیشنهادی در شبکه‌های ابری (Cloud Networks)، اینترنت اشیا (IoT) و شبکه‌های بی‌سیم 5G قابل استفاده است. در تحقیقات آینده، می‌توان عملکرد این روش را در محیط‌های مختلف مورد بررسی قرار داد.
- ۴. استفاده از شبکه‌های عصبی بازگشتی (RNN) و تقویتی: شبکه‌های یادگیری تقویتی (Reinforcement Learning) به کنترلر کمک می‌کنند تا مسیرهای بهینه را نه تنها بر اساس وضعیت فعلی، بلکه بر اساس پیش‌بینی آینده نیز انتخاب کنند. این موضوع می‌تواند تأثیر بسزایی در کاهش تأخیر و بهبود کیفیت سرویس (QoS) داشته باشد (Myunghoon et al., ۲۰۲۰).
- به‌طور کلی، هوشمندسازی مسیریابی در SDN با استفاده از آماراتای یادگیر یک رویکرد انعطاف‌پذیر، تطبیق‌پذیر و کم‌هزینه برای بهبود کارایی شبکه است (Kreutz et al., ۲۰۱۵; Myunghoon et al., ۲۰۲۰). برخلاف روش‌های سنتی که بر مبنای اطلاعات ثابت و توپولوژی ایستا تصمیم‌گیری می‌کنند، روش پیشنهادی قادر است با استفاده از بازخوردگیری مداوم از شبکه، بهینه‌سازی پویای مسیرها را انجام دهد. این ویژگی باعث می‌شود که شبکه‌های SDN بتوانند به صورت هوشمندانه‌تر و کارآمدتر عمل کنند و در محیط‌های پرنوسان و پیچیده نیز عملکرد قابل قبولی داشته باشند.
- با گسترش فناوری‌های ارتباطی مانند 5G، اینترنت اشیا (IoT) و رایانش ابری، استفاده از الگوریتم‌های هوشمند مسیریابی در SDN بیش از پیش اهمیت پیدا خواهد کرد. روش پیشنهادی در این مقاله می‌تواند پایه‌ای برای توسعه الگوریتم‌های پیشرفته‌تر در آینده



باشد که علاوه بر کاهش تأخیر و افزایش بهره‌وری شبکه، بتوانند به‌صورت خودمختار و بدون نیاز به مداخله انسانی تصمیم‌گیری کنند (Kim and Feamster, ۲۰۱۳; Kreutz et al., ۲۰۱۵).

## ۶- مراجع

- [۱] J. Calle-Cancho, C. Cruz-Carrasco, D. Cortés-Polo, J. Galeano-Brajones, and J. Carmona-Murillo Enhancing Programmability in Next-Generation Networks: An Innovative Simulation Approach Electronics, ۱۳(۵۳۲), ۲۰۲۴, doi: ۱۰.۳۳۹۰/electronics۱۳۰۳۰۵۳۲
- [۲] H. Kim and N. Feamster  
Improving network management with software defined networking  
IEEE Communications Magazine, vol. ۵۱, no. ۲, ۲۰۱۳, pp. ۱۱۴–۱۱۹
- [۳] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig  
Software-defined networking: A comprehensive survey  
Proceedings of the IEEE, vol. ۱۰۳, no. ۱, ۲۰۱۵, pp. ۱۴–۷۶
- [۴] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner  
OpenFlow: Enabling innovation in campus networks  
ACM SIGCOMM Computer Communication Review, vol. ۳۸, no. ۲, ۲۰۰۸, pp. ۶۹–۷۴
- [۵] D. L. Tennenhouse and D. J. Wetherall  
Towards an active network architecture  
Computer Communication Review, vol. ۲۶, no. ۲, ۱۹۹۶, pp. ۵–۱۸
- [۶] R. Alvizu, G. Maier, N. Kukreja, A. Pattavina, R. Morro, A. Capello, and C. Cavazzoni  
Comprehensive survey on t-sdn: Software-defined networking for transport networks  
IEEE Communications Surveys & Tutorials, vol. ۱۹, no. ۴, ۲۰۱۷, pp. ۲۲۳۲–۲۲۸۳
- [۷] K. Nisar, E. R. Jimson, I. Welch, R. Hassan, A. H. Mohd Aman, A. H. Sodhro, S. Pirbhulal, and S. Khan  
A survey on the architecture, application, and security of software defined networking: Challenges and open issues  
Internet of Things, vol. ۱۲, ۲۰۲۰, p. ۱۰۰۲۸۹
- [۸] Y. C. Wang and T. J. Hsiao  
URBM: User-Rank-Based Management of Flows in Data Center Networks through SDN  
Proceedings of the ۴th International Conference on Computer Communication and the Internet (ICCCI), ۲۰۲۲, pp. ۱۴۲–۱۴۹
- [۹] J. Myunghoon, K. Namgi, J. Yehoon, and L. Byoung-Dai  
An Efficient Network Resource Management in SDN for Cloud Services  
Symmetry, vol. ۱۲, no. ۹, ۲۰۲۰, p. ۱۵۵۶



[۱۰] J. Son, T. He, and R. Buyya

CloudSimSDN-NFV: Modeling and Simulation of Network Function Virtualization and Service Function Chaining in Edge Computing Environments

Software: Practice and Experience, vol. ۴۹, no. ۱۲, ۲۰۱۹, pp. ۱۷۴۸–۱۷۶۴

## ۴- Conclusion

This article provides an overview of an intelligent routing approach for software-defined networks (SDN) based on learning automata. By dynamically adjusting path selections through continuous feedback from network conditions, the proposed method reduces latency, balances traffic more effectively, and utilizes bandwidth more efficiently than conventional static routing algorithms. Although the results show promise for improving overall network performance and flexibility, further research is needed to address large-scale deployment challenges and explore more advanced learning techniques. Ultimately, this study highlights how learning automata can contribute to more adaptable and efficient SDN environments, paving the way for future advancements in next-generation networking.