

Analisis Perbandingan Performansi Strategi Forwarding Multicast dan Self-Learning pada Jaringan Vehicle-to-Vehicle Berbasis Named Data Networking

Seshariana Rahma Melati^{*1}, Lia Hafiza², Putri Rahmawati³, Aisyah Novfitri⁴

^{1,2,3,4} Program Studi S1 Teknik Telekomunikasi, Universitas Telkom, Jakarta, Indonesia
Email: ¹sesharianarahmam@telkomuniversity.ac.id, ²liahfza@telkomuniversity.ac.id,
³putirrahmawati@telkomuniversity.ac.id, ⁴aisyahnovf@telkomuniversity.ac.id

Abstrak

Vehicular Ad-hoc Network (VANET) merupakan jaringan kendaraan dengan mobilitas tinggi dengan perubahan lokasi yang cepat dan koneksi jaringan yang tidak stabil. Named Data Networking (NDN) merupakan arsitektur internet masa depan yang berpusat pada data (data-centric) dan menyediakan mekanisme forwarding yang bersifat stateful, sehingga sesuai untuk mendukung karakteristik VANET. Pada arsitektur NDN, strategi forwarding berperan penting dalam menentukan efisiensi pengiriman paket. Penelitian ini bertujuan membandingkan performansi dua strategi forwarding pada jaringan VANET berbasis NDN, yaitu strategi multicast-vanet yang berbasis broadcast dan strategi self-learning forwarding yang mampu mempelajari rute secara mandiri tanpa memerlukan protokol routing. Pengujian dilakukan menggunakan simulator ndnSIM yang diintegrasikan dengan Simulation of Urban MObility (SUMO) melalui platform NDN4IVC, dengan skenario jalur kendaraan berbentuk peta grid dan variasi jumlah node kendaraan, serta mengukur tiga parameter performansi, yaitu Cache Hit Ratio, Throughput, dan Round-Trip Time (RTT). Hasil pengujian menunjukkan bahwa strategi self-learning memperoleh rata-rata Cache Hit Ratio lebih tinggi dibandingkan multicast-vanet, sedangkan Throughput kedua strategi hampir identik. Pada parameter RTT, self-learning menghasilkan nilai lebih rendah pada sebagian besar skenario. Hal ini terjadi karena self-learning mempelajari jalur dari mekanisme broadcast dan menetapkan jalur unicast setelah tahap penemuan awal, sehingga pemanfaatan cache lebih efektif dan waktu pertukaran data lebih singkat. Dengan demikian, strategi self-learning forwarding memberikan efisiensi yang lebih baik dibandingkan multicast-vanet pada jaringan VANET-NDN dan dapat menjadi acuan empiris dalam pemilihan strategi forwarding untuk pengembangan jaringan kendaraan berbasis Named Data Networking.

Kata kunci: *multicast, NDN, self-learning forwarding, strategi forwarding, VANET*

Abstract

Vehicular Ad-hoc Network (VANET) is a vehicular network with high mobility, causing rapid location changes and unstable network connections. Named Data Networking (NDN) is a data-centric future Internet architecture that provides a stateful forwarding mechanism, making it suitable for supporting VANET characteristics. In NDN, the forwarding strategy plays an important role in determining packet delivery efficiency. This study aims to compare the performance of two forwarding strategies on an NDN-based VANET, namely the broadcast-based multicast-vanet strategy and the self-learning forwarding strategy, which learns routes independently without requiring a routing protocol. The evaluation used the ndnSIM simulator integrated with Simulation of Urban MObility (SUMO) through the NDN4IVC platform, with a grid-map vehicle route scenario and varying numbers of vehicle nodes, measuring three parameters: Cache Hit Ratio, Throughput, and Round-Trip Time (RTT). Results show that the self-learning strategy achieves a higher average Cache Hit Ratio than multicast-vanet, while the Throughput of both strategies is almost identical. For the RTT parameter, self-learning produces lower values in most scenario. This occurs because self-learning learns paths from the broadcast mechanism and establishes a unicast path after the initial discovery stage, resulting in more effective cache utilization and shorter data exchange time. Therefore, the self-learning forwarding strategy provides better efficiency than multicast-vanet on VANET-NDN networks and contributes as an empirical reference for selecting forwarding strategies in developing vehicular networks based on Named Data Networking.

Keywords: *forwarding strategy, multicast, NDN, self-learning forwarding, VANET*

1. PENDAHULUAN

Pertukaran data informasi pada Internet saat ini masih menggunakan arsitektur jaringan *Internet Protocol* (IP). Arsitektur jaringan IP memberikan identitas pada data berupa alamat sumber (IP *source*) dan alamat tujuan (IP *destination*), sehingga hal terpenting pada jaringan IP adalah mengetahui alamat IP sumber dan tujuan, tanpa mempertimbangkan seberapa jauh letak perangkat dengan alamat IP tujuan tersebut. Sebagian besar model pengiriman data memprioritaskan data apa yang dibutuhkan tanpa mempertimbangkan lokasi *server* dan *host*, padahal semakin jauh jarak antara *server* dan *host* akan memengaruhi kualitas *delay* yang dihasilkan.

Named Data Networking (NDN), sebagai salah satu rancangan arsitektur internet masa depan yang sedang dikembangkan, mengubah sudut pandang jaringan yang sebelumnya berpusat pada host (host-centric) menjadi berpusat pada data (data-centric) (Jacobson et al., 2009; Zhang et al., 2014). Jika pada jaringan IP setiap perangkat harus mengetahui alamat IP dari penyedia konten data (server) dan harus menjangkau alamat IP tersebut, maka pada NDN pengguna tidak harus mengetahui siapa penyedia konten data, melainkan cukup memperoleh konten dari perangkat mana pun yang dekat dengan pengguna dan memiliki konten yang diminta. Hal ini dimungkinkan karena pada NDN setiap router dilengkapi dengan sistem caching untuk menyimpan salinan data dari penyedia konten (producer) serta mekanisme forwarding untuk transmisi data, sehingga NDN mampu mempercepat pertukaran informasi secara efisien (Saxena et al., 2016; Xylomenos et al., 2014).

Meskipun NDN pada awalnya dirancang untuk memperbaiki struktur komunikasi Internet, saat ini sejumlah penelitian tengah mengkaji perluasan NDN pada jaringan nirkabel, seperti Vehicular Ad-hoc Networks (VANETs) (Amadeo et al., 2016; Khelifi et al., 2020). Secara alami, NDN mampu memberikan banyak manfaat, mulai dari dukungan mobilitas hingga kemudahan konfigurasi dengan biaya (cost) yang rendah. VANET merupakan jaringan khusus yang menyediakan komunikasi nirkabel bagi kendaraan yang bergerak dan infrastruktur lainnya guna mengurangi angka kecelakaan serta mengumpulkan berbagai informasi terkait lingkungan kendaraan yang berguna sebagai penunjang navigasi pengemudi untuk meningkatkan keselamatan pengemudi dan penumpang. Arsitektur Internet tradisional, yaitu TCP/IP, menyediakan koneksi end-to-end yang stabil antara klien dan server untuk mengirimkan informasi. Namun, pada lingkungan kendaraan koneksi ini menjadi tidak stabil akibat mobilitas kendaraan yang mengakibatkan perubahan lokasi, baik pada klien, server, maupun keduanya (Grassi et al., 2014). Untuk mengatasi situasi tersebut, NDN diusulkan sebagai arsitektur baru untuk pengambilan data dan dukungan mobilitas pada VANET.

Paradigma baru ini meningkatkan akses dan penyebaran konten dengan memisahkan konten dari lokasi asalnya (producer) (Ahed et al., 2022). Tujuan utama dari aspek forwarding dan routing pada jaringan NDN adalah mengatasi inefisiensi perutean yang terdapat pada jaringan berbasis IP saat ini serta memastikan penyebaran data yang efisien, khususnya pada lingkungan kendaraan. Operasi forwarding yang bersifat stateful dan adaptif memberikan manfaat yang besar bagi komunikasi kendaraan (Yi et al., 2012; Yi et al., 2013). Untuk menjawab tantangan VANET dalam hal ketersediaan tautan (link) pada proses transmisi data, diperlukan skema strategi forwarding yang mampu mendukung jaringan VANET dengan memanfaatkan arsitektur jaringan NDN.

Terdapat beberapa pendekatan strategi forwarding yang dapat digunakan pada jaringan VANET berbasis NDN. Salah satunya adalah strategi berbasis broadcast atau multicast, yang meneruskan paket interest ke seluruh antarmuka (face) yang tersedia. Strategi ini menawarkan keandalan dalam proses penemuan (discovery) konten karena paket dikirim ke banyak jalur sekaligus, namun berpotensi menimbulkan overhead yang tinggi akibat duplikasi paket (Kuai et al., 2014). Sebagai alternatif, self-learning forwarding menawarkan pendekatan yang lebih hemat, yaitu dengan melakukan broadcast paket interest hanya pada tahap awal untuk menemukan jalur, kemudian mempelajari rute yang berhasil berdasarkan paket data yang kembali, sehingga paket-paket berikutnya dapat dikirim secara unicast tanpa memerlukan protokol perutean (Shi et al., 2017).

Meskipun demikian, kedua pendekatan tersebut masih memiliki keterbatasan yang dilaporkan pada penelitian terdahulu. Strategi berbasis broadcast seperti multicast-vanet rentan mengalami ledakan paket (broadcast storm) dan duplikasi transmisi ketika kepadatan kendaraan meningkat, sehingga membebani kanal nirkabel yang kapasitasnya terbatas (Heidari et al., 2025; Kuai et al., 2014; Ngo et al., 2023). Di

sisi lain, jalur yang telah dipelajari oleh self-learning dapat menjadi usang (stale) akibat perubahan topologi yang cepat pada lingkungan kendaraan, sehingga efektivitasnya pada jaringan dengan mobilitas tinggi masih perlu dibuktikan secara empiris (Shi et al., 2017; Wang et al., 2023). Sebagian besar penelitian terdahulu berfokus pada pengembangan konsep atau penyempurnaan satu strategi tertentu, sedangkan kajian yang membandingkan secara langsung performansi strategi multicast-vanet dan self-learning pada kondisi pengujian yang sama di jaringan VANET-NDN masih terbatas. Ketiadaan perbandingan langsung tersebut menjadi celah penelitian yang diisi dalam penelitian ini.

Kontribusi utama penelitian ini adalah menganalisis dan membandingkan performansi kedua strategi forwarding tersebut, yaitu multicast-vanet dan self-learning forwarding, pada jaringan VANET berbasis NDN. Perbandingan dilakukan terhadap tiga parameter performansi, yaitu Cache Hit Ratio, Throughput, dan Round-Trip Time (RTT), dengan variasi kepadatan jaringan (jumlah node) pada skenario jalur kendaraan berbentuk peta grid. Melalui analisis ini, diharapkan dapat diperoleh pemahaman yang lebih mendalam mengenai karakteristik dan keunggulan masing-masing strategi forwarding, sehingga dapat menjadi acuan dalam pemilihan strategi forwarding yang sesuai untuk jaringan VANET-NDN.

1.1. Penelitian Terkait

Penelitian mengenai forwarding pada NDN telah berkembang pesat sejak arsitektur ini pertama kali diperkenalkan. (Yi et al., 2012) memperkenalkan konsep adaptive forwarding yang memungkinkan router NDN mendeteksi permasalahan jaringan melalui pengamatan lalu lintas dua arah paket Interest dan Data serta menjelajahi berbagai jalur alternatif tanpa membentuk loop. Konsep ini diperkuat oleh gagasan stateful forwarding plane (Yi et al., 2013), yang menegaskan bahwa kemampuan router NDN dalam menyimpan status Interest yang tertunda (pending) menjadi dasar bagi strategi forwarding yang cerdas dan adaptif. Kedua konsep tersebut menjadi landasan bagi pengembangan berbagai strategi forwarding, termasuk strategi multicast dan self-learning.

Pada penelitian (Shi et al., 2017), mekanisme self-learning berbasis broadcast memungkinkan jaringan beradaptasi terhadap lingkungan yang berfluktuasi dan mendukung mobilitas tanpa menggunakan protokol perutean atau protokol kendali lainnya. Keunggulan ini terutama tampak pada skenario jaringan mobile ad hoc dan nirkabel yang tidak memiliki infrastruktur stabil, yang apabila menggunakan pengumuman perutean berulang akan membutuhkan waktu komputasi dan energi yang berlebihan. Karena self-learning tidak membutuhkan pengetahuan awal mengenai lokasi data, mekanisme ini bekerja dengan baik pada topologi jaringan yang berpusat pada data seperti NDN. Hasil emulasi pada penelitian tersebut menunjukkan bahwa self-learning NDN mentransmisikan lebih sedikit paket dibandingkan pendekatan broadcast dan Ethernet, meskipun setiap teknik forwarding pada akhirnya dapat mengidentifikasi hilangnya koneksi dan berpindah ke jalur yang berbeda dengan pemanfaatan bandwidth dan waktu pemulihan yang berbeda secara signifikan.

Selanjutnya, penelitian (Liang et al., 2020) menyempurnakan self-learning agar dapat bekerja secara out-of-the-box pada jaringan edge. Metode self-learning terdahulu memiliki beberapa kekurangan, yaitu hanya dapat mempelajari satu jalur meskipun terdapat banyak jalur atau sumber data, tidak dapat membedakan antar-face yang memiliki perilaku berbeda (misalnya perilaku multicast Wi-Fi yang lebih buruk dibandingkan unicast), serta tidak memperhitungkan kasus delay tolerance yang memungkinkan face tidak tersedia. Penyempurnaan yang dilakukan memungkinkan metode ini menggunakan banyak jalur, mempelajari dan memilih unicast face setelah pertukaran Interest-Data melalui multicast face di Wi-Fi, serta menyimpan (buffer) interest dan meneruskannya ketika rute menjadi tersedia agar dapat berfungsi pada kondisi delay-tolerant.

Pada penelitian (Rahman & Zhang, 2021), self-learning terdahulu diubah menjadi teknik Data-centric Ad-hoc Forwarding (DAF) untuk membangun baseline NDN pada MANETs. DAF mengambil keseimbangan antara flooding dan self-learning, serta mengandalkan mekanisme umpan balik-reaksi di dalam jaringan (in-network feedback-reaction) alih-alih broadcast berbasis timeout konsumen. DAF membentuk dan memperbarui entri FIB melalui umpan balik positif dan negatif sehingga mengurangi tumbukan (collision) paket dan pesan protokol, serta merespons kegagalan secara terdesentralisasi dan

oportunistik. Pendekatan ini terinspirasi oleh Ad-hoc On-Demand Distance Vector (AODV) pada MANETs berbasis IP, namun bekerja pada domain berbasis nama (name-based).

Dalam konteks VANET, (Grassi et al., 2014) memperkenalkan V-NDN sebagai implementasi awal NDN untuk jaringan kendaraan dan menunjukkan potensi NDN sebagai arsitektur pemersatu bagi berbagai perangkat komputasi, baik yang terhubung melalui infrastruktur, ad hoc, maupun jaringan intermittent. Penelitian tersebut kemudian dilanjutkan dengan Navigo (Grassi et al., 2015), yang mengusulkan mekanisme forwarding interest berbasis geolokasi untuk pengambilan konten pada komunikasi V2V dan V2I. (Amadeo et al., 2012) mengusulkan CRoWN sebagai arsitektur content-centric untuk VANET, sementara (Wang et al., 2012) menunjukkan pemanfaatan named data untuk penyebaran informasi lalu lintas secara cepat. (Kuai et al., 2014) secara khusus mengevaluasi kinerja interest broadcast pada vehicular NDN dan menyoroti persoalan overhead akibat duplikasi paket pada strategi berbasis broadcast.

Sejumlah penelitian dalam lima tahun terakhir menegaskan bahwa strategi forwarding dan caching masih menjadi isu utama pada vehicular NDN. (Al-Qutwani et al., 2021) mengusulkan mekanisme request/advertise (RACF) yang memanfaatkan parameter pergerakan kendaraan untuk membangun jalur forwarding dinamis sehingga menekan overhead dan latensi. (Wang et al., 2023) mengembangkan strategi forwarding prediktif berbasis prediksi posisi kendaraan tetangga yang mampu mengurangi jumlah paket yang diteruskan lebih dari 21%, sementara (Khan & Lim, 2024) memanfaatkan prediksi mobilitas recursive least squares untuk menekan latensi pengiriman data, dan (Khan et al., 2025) mengusulkan skema penerusan adaptif untuk layanan infotainment. (Wang & Hou, 2024) secara khusus menangani persoalan broadcast storm pada vehicular NDN melalui penundaan rebroadcast dan identifikasi visual antar-kendaraan. Pada aspek pemanfaatan cache, (Melati et al., 2021) menunjukkan bahwa penambahan layanan penamaan NDNS mampu meningkatkan cache hit ratio pada berbagai topologi, (Amadeo et al., 2021) dan (Chen et al., 2022) membuktikan bahwa keputusan caching yang mempertimbangkan popularitas serta karakteristik ruang-waktu konten dapat meningkatkan cache hit ratio dan menekan waktu pengambilan konten, sedangkan (Liao et al., 2024) mengusulkan caching adaptif terhadap dinamika lingkungan (EADC) yang memperbaiki delay dan cache hit ratio. Dukungan mobilitas producer juga telah dikaji secara komparatif oleh (Alkwai et al., 2022). Perkembangan tersebut menunjukkan bahwa efisiensi penerusan paket dan pemanfaatan cache merupakan dua aspek yang saling terkait dalam perancangan strategi forwarding pada jaringan kendaraan berbasis NDN.

Berdasarkan tinjauan tersebut, sebagian besar penelitian sebelumnya berfokus pada pengembangan atau penyempurnaan satu strategi forwarding tertentu. Penelitian ini melengkapi kajian tersebut dengan menyajikan analisis perbandingan langsung antara strategi multicast-vanet dan self-learning forwarding pada jaringan VANET-NDN, sehingga karakteristik masing-masing strategi dapat dievaluasi secara berdampingan pada kondisi pengujian yang sama. Ringkasan penelitian terdahulu beserta metode, hasil utama, dan celah penelitian yang ingin diisi oleh penelitian ini disajikan pada Tabel 1.

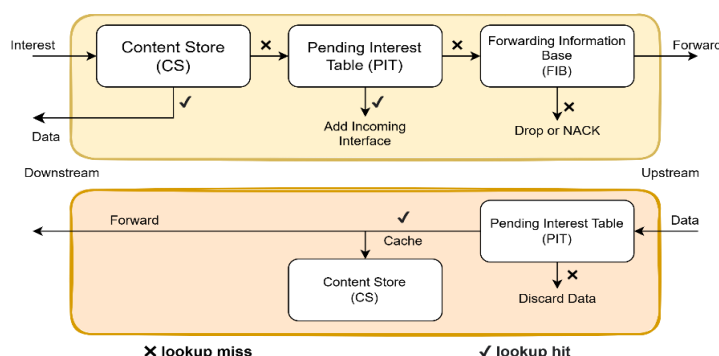
2. METODE PENELITIAN

2.1. Named Data Networking (NDN)

Forwarding pada arsitektur jaringan NDN berfokus pada penerusan konten data dibandingkan pada identitas host pengirim dan penerima sebagaimana pada arsitektur jaringan Internet Protocol (IP) saat ini (Jacobson et al., 2009; Zhang et al., 2014). Pada NDN, pengguna tidak harus meminta data dari server tertentu, melainkan meminta konten dari jaringan, dan jaringan akan menentukan node mana yang akan merespons permintaan tersebut. Dengan demikian, permintaan konten tidak hanya dapat dilayani oleh server tertentu, tetapi juga oleh node mana pun di dalam jaringan yang memiliki konten yang diminta oleh konsumen. Gambar 1 mengilustrasikan mekanisme forwarding pada jaringan NDN secara umum, yang dimulai ketika konsumen meminta konten dari jaringan dengan mengirimkan paket interest.

Tabel 1. Ringkasan Penelitian Terdahulu dan Celah Penelitian

Penelitian	Metode/Fokus	Hasil Utama	Celah Penelitian
Yi et al. (2012, 2013)	Konsep <i>adaptive forwarding</i> dan <i>stateful forwarding plane</i> pada NDN	Router NDN dapat mendeteksi masalah jaringan dan menjelajahi jalur alternatif tanpa <i>loop</i>	Belum diterapkan dan diuji pada lingkungan kendaraan
Kuai et al. (2014)	Evaluasi <i>interest broadcast</i> pada <i>vehicular</i> NDN	<i>Broadcast</i> andal untuk penemuan konten, tetapi menimbulkan <i>overhead</i> duplikasi paket	Tidak membandingkan dengan strategi pembelajaran jalur
Grassi et al. (2014, 2015)	Implementasi V-NDN dan <i>forwarding</i> berbasis geolokasi (Navigo)	NDN layak sebagai arsitektur komunikasi V2V/V2I	Memerlukan informasi geolokasi; belum mengkaji <i>self-learning</i>
Shi et al. (2017)	Perancangan <i>self-learning forwarding</i> berbasis <i>broadcast</i>	<i>Self-learning</i> mentransmisikan paket lebih sedikit daripada <i>broadcast</i> murni	Evaluasi pada emulasi jaringan statis, bukan skenario VANET
Liang et al. (2020)	Penyempurnaan <i>self-learning</i> pada jaringan <i>edge</i>	Mendukung banyak jalur dan pemilihan <i>unicast face</i> pada Wi-Fi	Belum dievaluasi pada mobilitas kendaraan
Rahman & Zhang (2021)	<i>Data-centric Ad-hoc Forwarding</i> (DAF) sebagai <i>baseline</i> NDN pada MANET	Umpan balik dalam jaringan mengurangi tumbukan paket dan pesan protokol	Fokus pada MANET umum, bukan karakteristik VANET
Al-Qutwani et al. (2021); Wang et al. (2023); Khan & Lim (2024)	Strategi <i>forwarding</i> VNDN berbasis parameter gerak dan prediksi mobilitas	Menekan <i>overhead</i> , latensi, dan jumlah paket yang diteruskan	Skema baru berkompleksitas tinggi; tidak membandingkan strategi bawaan NFD
Penelitian ini	Perbandingan langsung <i>multicast-vanet</i> dan <i>self-learning</i> pada VANET-NDN (ndnSIM + SUMO)	Analisis <i>Cache Hit Ratio</i> , <i>Throughput</i> , dan RTT pada variasi 30-120 <i>node</i>	Mengisi celah perbandingan empiris antar-strategi pada kondisi uji yang sama



Gambar 1. Mekanisme *Forwarding* pada NDN (Saxena et al., 2016)

Ketika router NDN menerima paket interest, router akan memeriksa apakah paket tersebut terdapat di dalam Content Store (CS) atau tidak. Jika terdapat paket data yang sesuai, data akan langsung dikirimkan kembali ke antarmuka konsumen yang meminta konten. Jika tidak ditemukan paket data

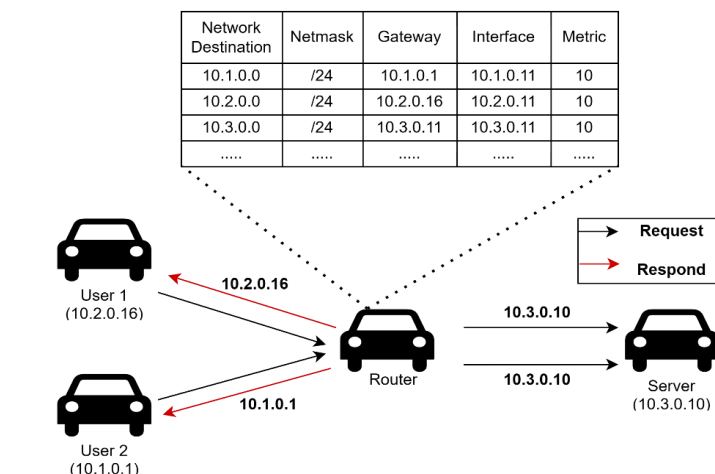
yang sesuai pada CS, paket interest diperiksa kembali pada Pending Interest Table (PIT), yaitu bagian yang menyimpan paket interest yang telah diteruskan dan sedang menunggu paket data kembali. Jika ditemukan entri yang sesuai pada PIT, router akan menambahkan antarmuka masuk (incoming interface) pada entri tersebut. Apabila paket interest tidak terdapat pada PIT, paket akan ditambahkan sebagai entri baru pada PIT dan kemudian diteruskan ke antarmuka yang dipilih oleh modul strategi forwarding melalui Forwarding Information Base (FIB) (Saxena et al., 2016).

Paket data yang dikembalikan ke antarmuka konsumen selalu menempuh jalur yang berlawanan dengan paket interest yang masuk. Dalam pertukaran paket pada jaringan NDN, baik paket interest maupun paket data tidak membawa alamat host konsumen maupun producer; router meneruskan paket interest ke producer berdasarkan nama yang dibawa dalam paket, dan meneruskan paket data ke konsumen berdasarkan informasi status pada PIT di setiap hop. Simetri pertukaran paket Interest dan Data ini memicu loop kendali hop-by-hop dan menghilangkan kebutuhan akan konsep node sumber atau tujuan dalam transmisi data sebagaimana pada arsitektur jaringan IP (Zhang et al., 2014).

Modul strategi forwarding yang terdapat pada setiap node merupakan kunci ketahanan (resilience) dan efisiensi NDN (Yi et al., 2012; Yi et al., 2013). Melalui pemanfaatan kemampuan multipath NDN secara efektif, strategi forwarding yang adaptif dapat mengirimkan paket interest sepanjang jalur dengan kinerja terbaik, menghindari bottleneck dan kegagalan, serta menyeimbangkan beban (load) di seluruh jalur. Oleh karena itu, penelitian mengenai strategi forwarding menjadi penting untuk merancang strategi yang sederhana dan efektif bagi berbagai konteks dan perangkat pada jaringan berbasis NDN.

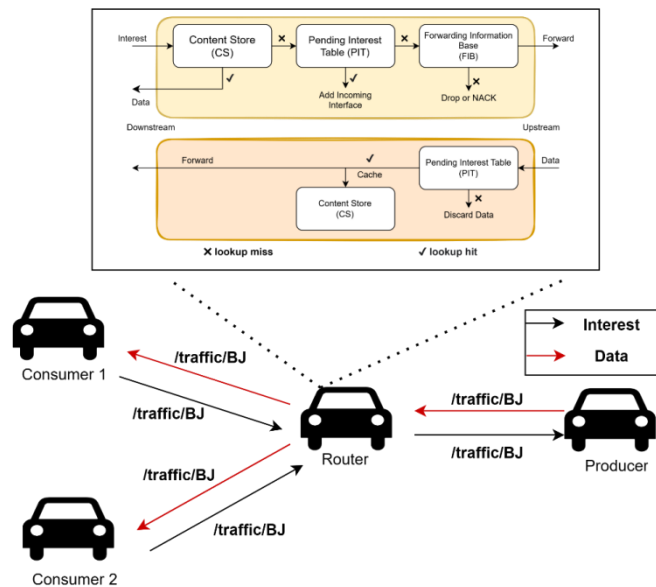
2.2. VANET pada Named Data Networking (NDN)

Proses pengiriman data pada arsitektur VANET yang menggunakan arsitektur jaringan IP berfokus pada tabel perutean (routing table) serta proses pembaruan tautan (link). Perubahan lokasi kendaraan yang cepat menjadi hambatan dalam penggunaan arsitektur ini, mengingat IP merupakan jaringan yang berpusat pada host (host-centric) yang transmisi datanya berfokus pada alamat IP, sehingga mengharuskan pengiriman data hanya pada alamat IP tujuan dan alamat IP sumber (Khelifi et al., 2020). Arsitektur VANET pada jaringan IP ditunjukkan pada Gambar 2.



Gambar 2. Arsitektur VANET pada Jaringan IP (Yi et al., 2013)

Sementara itu, pada arsitektur VANET yang menggunakan arsitektur jaringan NDN sebagaimana ditunjukkan pada Gambar 3, proses pengiriman data dapat dilakukan oleh perangkat jaringan mana pun yang memiliki konten yang diminta oleh konsumen (Amadeo et al., 2016; Grassi et al., 2014). Berbeda dengan IP yang berfokus pada tabel perutean antara alamat sumber dan tujuan, NDN memiliki mekanisme forwarding yang bersifat stateful, sehingga dapat mempercepat proses pertukaran data pada lingkungan kendaraan yang dinamis (Chen et al., 2014).



Gambar 3. Arsitektur VANET pada Jaringan NDN (Yi et al., 2013)

2.3. Strategi Forwarding Multicast dan Self-Learning

Penelitian ini membandingkan dua strategi forwarding pada jaringan VANET-NDN, yaitu strategi multicast (multicast-vanet) dan strategi self-learning. Kedua strategi tersebut memiliki mekanisme kerja yang berbeda dalam menentukan bagaimana paket interest diteruskan ke seluruh jaringan.

Strategi multicast-vanet bekerja dengan meneruskan setiap paket interest ke seluruh antarmuka (face) yang tercantum pada FIB secara bersamaan. Dengan kata lain, strategi ini mengandalkan mekanisme broadcast atau flooding untuk menyebarkan permintaan konten ke seluruh jaringan. Kelebihan strategi ini adalah tingginya keandalan dalam proses penemuan (discovery) konten, karena paket interest dikirim melalui banyak jalur secara paralel sehingga peluang menemukan producer atau cache yang sesuai menjadi lebih besar. Namun, strategi multicast memiliki kelemahan berupa overhead yang tinggi akibat duplikasi paket pada jaringan, yang dapat meningkat seiring dengan bertambahnya kepadatan node (Kuai et al., 2014).

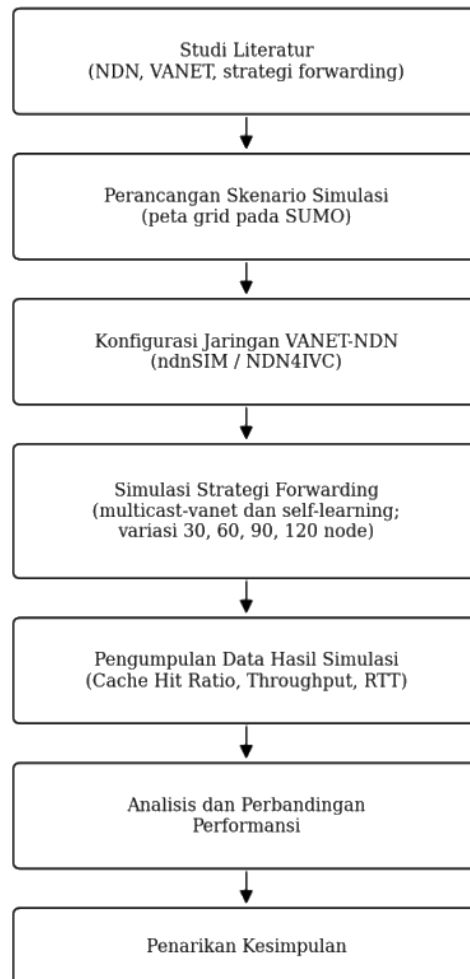
Strategi self-learning dirancang untuk mengurangi overhead tersebut dengan cara mempelajari rute secara mandiri. Pada tahap awal, ketika jalur menuju konten belum diketahui, self-learning melakukan broadcast paket interest ke seluruh antarmuka. Ketika paket data berhasil kembali, self-learning mencatat jalur yang berhasil tersebut ke dalam entri FIB, sehingga paket-paket interest berikutnya dengan prefix yang sama dapat diteruskan secara unicast melalui jalur yang telah diketahui, tanpa perlu melakukan flooding kembali (Shi et al., 2017). Dengan demikian, self-learning memungkinkan jaringan untuk membangun rute secara dinamis berdasarkan pengalaman pengiriman paket yang berhasil, tanpa memerlukan protokol perutean tambahan. Pendekatan ini sejalan dengan konsep adaptive forwarding (Yi et al., 2012) dan stateful forwarding plane (Yi et al., 2013) yang menjadi ciri khas arsitektur NDN.

Perbedaan mendasar antara kedua strategi ini terletak pada cara pemanfaatan jalur. Strategi multicast secara konsisten menyebarkan paket ke seluruh jalur pada setiap permintaan, sedangkan self-learning hanya melakukan penyebaran pada tahap penemuan awal dan selanjutnya memanfaatkan jalur yang telah dipelajari. Perbedaan mekanisme inilah yang menjadi fokus analisis pada penelitian ini, khususnya pengaruhnya terhadap nilai Cache Hit Ratio, Throughput, dan Round-Trip Time (RTT).

2.4. Tahapan Penelitian

Tahapan penelitian secara keseluruhan ditunjukkan pada Gambar 4. Penelitian diawali dengan studi literatur mengenai arsitektur NDN, karakteristik VANET, dan strategi forwarding. Tahap berikutnya adalah perancangan skenario simulasi berupa jalur kendaraan peta grid menggunakan SUMO, dilanjutkan dengan konfigurasi jaringan VANET-NDN pada ndnSIM melalui kerangka kerja

NDN4IVC. Simulasi kemudian dijalankan untuk kedua strategi forwarding, yaitu multicast-vanet dan self-learning, pada variasi jumlah node 30, 60, 90, dan 120. Data hasil simulasi berupa Cache Hit Ratio, Throughput, dan RTT selanjutnya dikumpulkan, dianalisis, dan dibandingkan untuk menarik kesimpulan.

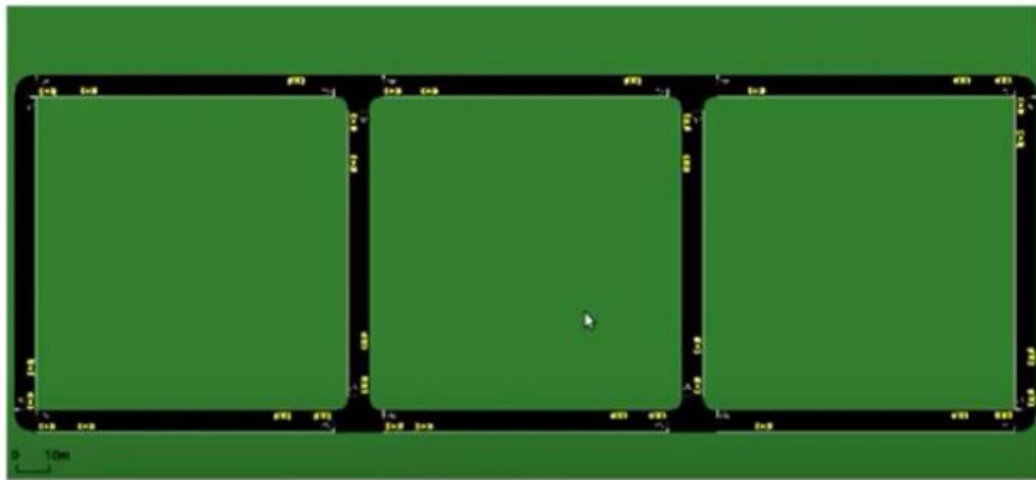


Gambar 4. Diagram Alur Penelitian

2.5. Perancangan Simulasi

Simulasi VANET-NDN dilakukan dengan skenario jalur kendaraan berbentuk peta grid (grid map) yang merepresentasikan wilayah pedesaan atau pegunungan. Pada wilayah tersebut terdapat jarak antara satu jalan dengan jalan lainnya karena adanya hambatan berupa perkebunan, sawah, atau permukiman penduduk. Perancangan jalur kendaraan divisualisasikan menggunakan Simulation of Urban MObility (SUMO) (Lopez et al., 2018), sedangkan proses simulasi jaringan NDN dijalankan menggunakan kerangka kerja yang mengintegrasikan ndnSIM (Afanasyev et al., 2012; Mastorakis et al., 2017) dengan lingkungan VANET (Araujo et al., 2022). Gambar 5 menunjukkan perancangan jalur kendaraan dengan peta grid yang digunakan pada penelitian ini.

Peta grid pada Gambar 5 terdiri atas tiga blok persegi yang dipisahkan oleh ruas jalan horizontal dan vertikal, sehingga membentuk pola jaringan jalan yang teratur menyerupai kisi (grid). Kendaraan yang direpresentasikan sebagai node bergerak di sepanjang ruas jalan yang mengelilingi setiap blok, baik searah maupun berlawanan arah, sebagaimana ditunjukkan oleh arah pergerakan node pada tiap ruas.



Gambar 5. Perancangan Jalur Kendaraan dengan Peta *Grid* (Shi et al., 2017)

Setiap kendaraan bertindak sebagai node NDN yang dapat berperan sebagai konsumen, producer, maupun router sekaligus, sehingga konten dapat diminta, disimpan pada cache, dan diteruskan oleh node mana pun yang berada di sepanjang lintasan. Struktur grid ini sengaja dipilih karena mampu merepresentasikan kondisi jaringan dengan jarak antar-jalan yang cukup lebar, sebagaimana lazim dijumpai pada wilayah pedesaan, sehingga konektivitas antar-node menjadi lebih menantang dibandingkan pada skenario jalan yang rapat. Dengan pola pergerakan yang mengikuti ruas jalan grid, jarak antar-kendaraan dan topologi jaringan berubah secara dinamis seiring waktu, sehingga strategi forwarding diuji pada kondisi mobilitas yang realistis. Skenario ini memungkinkan pengamatan bagaimana strategi multicast-vanet dan self-learning menanggapi perubahan kepadatan node, mulai dari 30 node yang merepresentasikan lalu lintas jarang hingga 120 node yang merepresentasikan lalu lintas padat. Parameter yang digunakan dalam simulasi ditampilkan pada Tabel 1.

Tabel 1. Parameter Simulasi Skenario

Parameter	Nilai
Jumlah <i>Node</i>	30; 60; 90; 120 <i>node</i>
Jumlah RSU	2 <i>node</i>
Strategi <i>Forwarding</i>	<i>Multicast-Vanet; Self-Learning</i>
Jarak antar-Kendaraan	10 meter
Interval <i>Interest</i>	1000 ms

Simulasi dijalankan pada komputer dengan prosesor Intel Core i5, memori RAM 8 GB, dan sistem operasi Ubuntu 20.04 LTS. Setiap skenario simulasi diulang sebanyak sepuluh kali dan nilai yang dilaporkan merupakan rata-rata dari seluruh pengulangan untuk memastikan konsistensi hasil pengukuran. Pemilihan rentang 30 hingga 120 node dimaksudkan untuk merepresentasikan kondisi lalu lintas dari jarang hingga padat pada wilayah dengan struktur jalan berjarak lebar, sehingga pengaruh kepadatan jaringan terhadap performansi masing-masing strategi forwarding dapat diamati secara bertahap.

2.6. Parameter Evaluasi Performansi

Performansi kedua strategi forwarding dievaluasi menggunakan tiga parameter, yaitu Cache Hit Ratio, Throughput, dan Round-Trip Time (RTT). Cache Hit Ratio menyatakan persentase permintaan konten yang dapat dilayani langsung oleh data yang tersimpan pada Content Store terhadap keseluruhan permintaan, sebagaimana dinyatakan pada Persamaan (1), dengan CHR menyatakan Cache Hit Ratio (%); semakin tinggi nilainya, semakin efektif pemanfaatan cache di dalam jaringan.

$$CHR = \frac{\text{Jumlah Cache Hit}}{\text{Jumlah Cache Hit} + \text{Jumlah Cache Miss}} \times 100\% \quad (1)$$

Throughput menyatakan laju data yang berhasil diterima oleh konsumen per satuan waktu, dihitung menggunakan Persamaan (2).

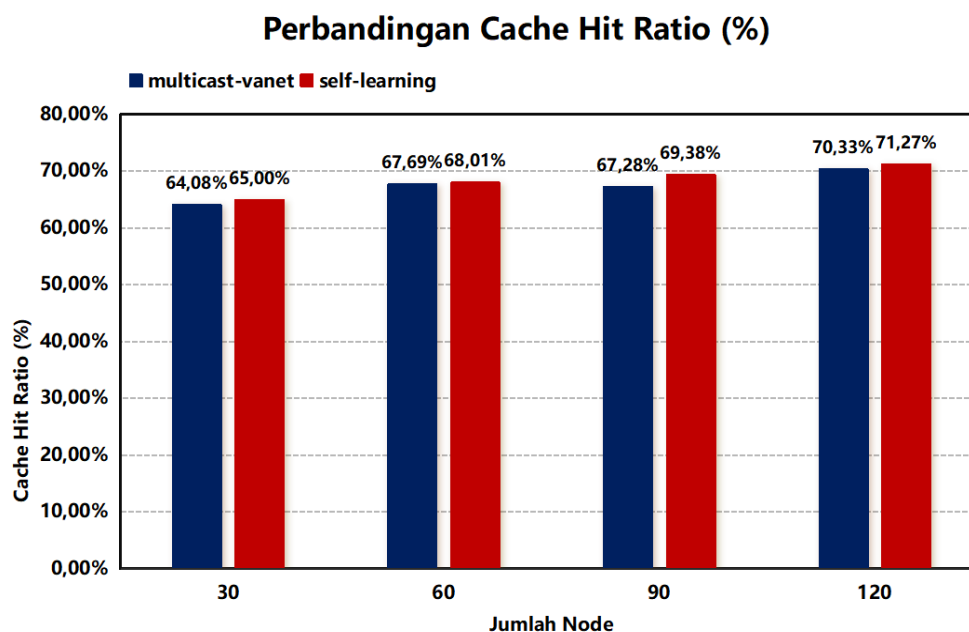
$$\text{Throughput} = \frac{\text{Jumlah Data Diterima (bit)}}{\text{Waktu Pengamatan (detik)}} \quad (2)$$

RTT menyatakan selang waktu antara pengiriman paket interest oleh konsumen hingga paket data yang bersesuaian diterima kembali, sebagaimana dinyatakan pada Persamaan (3), dengan t menyatakan waktu (ms). Nilai RTT yang dilaporkan merupakan rata-rata dari seluruh pertukaran paket yang berhasil selama simulasi berlangsung.

$$RTT = t_{\text{Data diterima}} - t_{\text{Interest dikirim}} \quad (3)$$

3. HASIL DAN PEMBAHASAN

3.1. Hasil Simulasi Perubahan Jumlah Node terhadap Cache Hit Ratio



Gambar 6. Grafik Hasil Simulasi Perubahan Jumlah *Node* terhadap *Cache Hit Ratio*

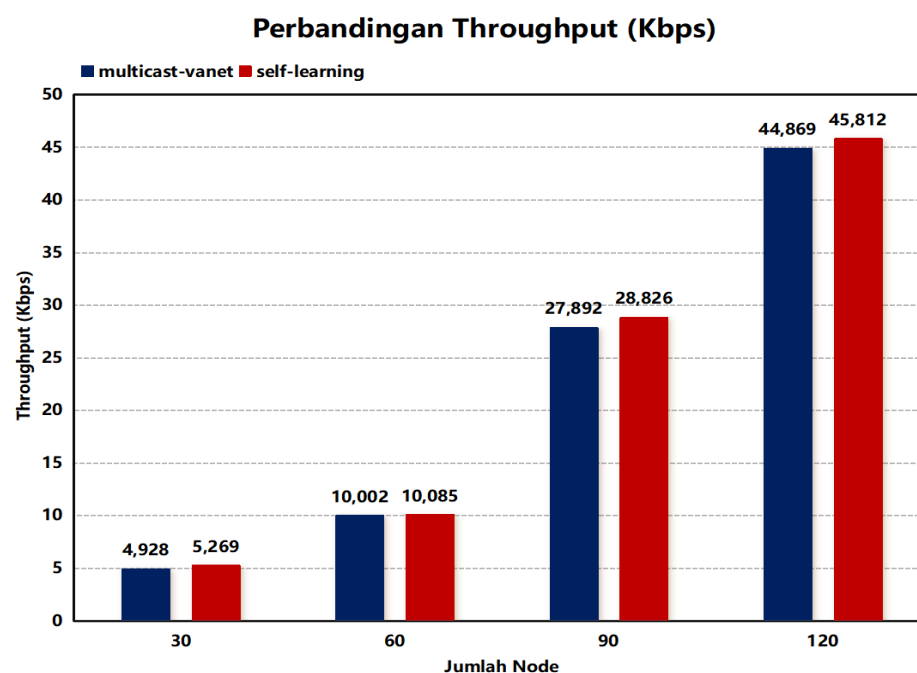
Berdasarkan grafik pada Gambar 6, nilai Cache Hit Ratio strategi self-learning secara konsisten lebih tinggi dibandingkan multicast-vanet pada seluruh variasi jumlah node. Pada jumlah 30 node, self-learning memperoleh nilai 65,00% sedangkan multicast-vanet 64,08%, sehingga self-learning lebih tinggi 1,44%. Pada jumlah 60 node, kedua strategi memperoleh nilai yang hampir sama, yaitu 68,01% untuk self-learning dan 67,69% untuk multicast-vanet, dengan selisih yang relatif kecil sebesar 0,47%. Pada jumlah 90 node, self-learning memperoleh 69,38% dibandingkan 67,28% pada multicast-vanet, sehingga self-learning lebih tinggi 3,12%. Selanjutnya, pada jumlah 120 node, self-learning memperoleh 71,27% dibandingkan 70,33% pada multicast-vanet, sehingga self-learning lebih tinggi 1,34%. Nilai Cache Hit Ratio kedua strategi juga terlihat cenderung meningkat seiring bertambahnya jumlah node. Rangkuman perbandingan nilai Cache Hit Ratio kedua strategi disajikan pada Tabel 2.

Tabel 2. Perbandingan Nilai Cache Hit Ratio Kedua Strategi.

Jumlah Node	CHR multicast-vanet	CHR self-learning	Keterangan
30	64,08%	65,00%	Self-learning lebih baik
60	67,69%	68,01%	Self-learning lebih baik
90	67,28%	69,38%	Self-learning lebih baik
120	70,33%	71,27%	Self-learning lebih baik

Dari grafik tersebut dapat disimpulkan bahwa strategi self-learning cenderung memiliki nilai Cache Hit Ratio yang lebih tinggi dibandingkan multicast-vanet. Hal ini terjadi karena pada self-learning, paket interest “discovery” dan “non-discovery” dapat dipenuhi dengan data yang tersimpan di dalam cache. Ketika merespons discovery of interest, NFD melampirkan pengumuman prefix (prefix announcement) pada Data, sehingga interest flooding dapat mempelajari jalur menuju cache alih-alih jalur menuju producer. Apabila interest berikutnya diteruskan ke cache tetapi tidak dapat dipenuhi oleh Data yang tersimpan, node cache akan meneruskan atau melakukan flooding terhadap interest tersebut apabila ditandai sebagai “discovery” (Shi et al., 2017). Mekanisme pembelajaran jalur menuju cache inilah yang membuat self-learning mampu memanfaatkan konten yang tersimpan secara lebih efektif dibandingkan strategi multicast yang murni berbasis broadcast.

3.2. Hasil Simulasi Perubahan Jumlah Node terhadap Nilai Throughput



Gambar 7. Grafik Hasil Simulasi Perubahan Jumlah *Node* terhadap *Throughput*

Berdasarkan grafik pada Gambar 7, nilai Throughput strategi self-learning secara konsisten sedikit lebih tinggi dibandingkan multicast-vanet pada seluruh variasi jumlah node. Pada jumlah 30 node, self-learning memperoleh 5,269 Kbps sedangkan multicast-vanet 4,928 Kbps, sehingga self-learning lebih tinggi 6,92%. Pada jumlah 60 node, kedua strategi memperoleh nilai yang hampir sama, yaitu 10,085 Kbps untuk self-learning dan 10,002 Kbps untuk multicast-vanet, dengan selisih yang relatif kecil sebesar 0,83%. Pada jumlah 90 node, self-learning memperoleh 28,826 Kbps dibandingkan 27,892 Kbps pada multicast-vanet, sehingga self-learning lebih tinggi 3,35%. Selanjutnya, pada jumlah 120 node, self-learning memperoleh 45,812 Kbps dibandingkan 44,869 Kbps pada multicast-vanet, sehingga self-learning lebih tinggi 2,10%. Nilai Throughput kedua strategi juga terlihat meningkat seiring

bertambahnya jumlah node. Rangkuman perbandingan nilai Throughput kedua strategi disajikan pada Tabel 3.

Tabel 3. Perbandingan Nilai Throughput Kedua Strategi.

Jumlah Node	Throughput multicast-vanet	Throughput self-learning	Keterangan
30	4,928 Kbps	5,269 Kbps	<i>Self-learning</i> lebih baik
60	10,002 Kbps	10,085 Kbps	<i>Self-learning</i> lebih baik
90	27,892 Kbps	28,826 Kbps	<i>Self-learning</i> lebih baik
120	44,869 Kbps	45,812 Kbps	<i>Self-learning</i> lebih baik

Dari grafik tersebut dapat disimpulkan bahwa strategi self-learning menghasilkan nilai Throughput yang praktis setara dengan multicast-vanet. Hal ini menunjukkan bahwa proses pembelajaran jalur pada self-learning tidak mengorbankan kapasitas pengiriman data. Kondisi ini terjadi karena self-learning tidak hanya menghilangkan protokol perutean, tetapi juga memungkinkan NDN memanfaatkan bandwidth pada seluruh tautan yang tersedia serta memastikan bahwa flooding pada tahap penemuan akan selalu mempelajari jalur tercepat. Ketika dua salinan paket interest tiba pada node yang sama melalui dua jalur berbeda selama flooding, salinan yang diterima kemudian dibuang (dropped), sehingga data balasan hanya dikembalikan melalui jalur tercepat. Setelah interest flooding, NFD menyisipkan entri FIB untuk prefix yang tercantum pada pengumuman prefix Data yang dikembalikan, dengan next-hop berupa node bertetangga tempat Data diterima. Jumlah entri FIB dinamis ini memiliki batas kapasitas; ketika FIB penuh, entri yang paling jarang digunakan akan dihapus menggunakan algoritma caching LRU (Least Recently Used) (Shi et al., 2017). Dengan mekanisme ini, self-learning mampu mempertahankan Throughput yang setara dengan multicast sambil menekan overhead pengiriman.

3.3. Hasil Simulasi Perubahan Jumlah Node terhadap Nilai RTT

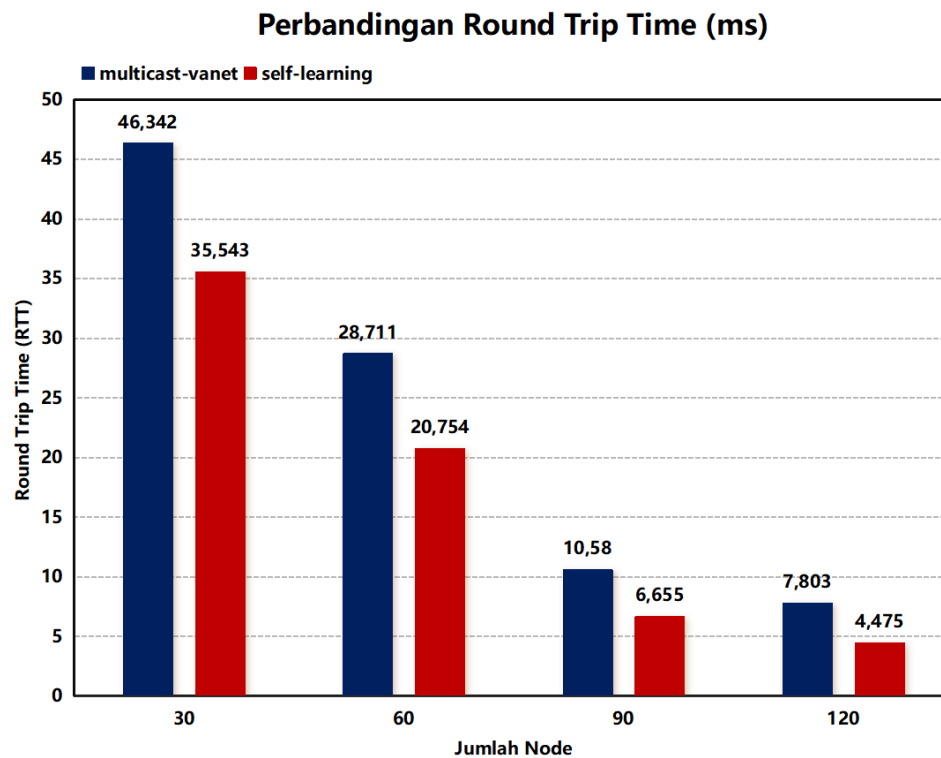
Berdasarkan grafik pada Gambar 8, strategi self-learning secara konsisten menghasilkan nilai RTT yang lebih rendah dibandingkan multicast-vanet pada seluruh variasi jumlah node. Pada jumlah 30 node, self-learning memperoleh 4,475 ms dibandingkan 7,803 ms pada multicast-vanet, sehingga self-learning menghasilkan RTT 42,65% lebih rendah dan menjadi penurunan relatif yang paling signifikan. Pada jumlah 60 node, self-learning memperoleh 6,655 ms dibandingkan 10,58 ms pada multicast-vanet, sehingga self-learning 37,10% lebih rendah. Pada jumlah 90 node, self-learning memperoleh 20,754 ms dibandingkan 28,711 ms pada multicast-vanet, sehingga self-learning 27,71% lebih rendah. Selanjutnya, pada jumlah 120 node, self-learning memperoleh 35,543 ms dibandingkan 46,342 ms pada multicast-vanet, sehingga self-learning 23,30% lebih rendah dengan selisih absolut terbesar, yaitu mencapai 10,799 ms. Nilai RTT kedua strategi juga terlihat semakin besar seiring bertambahnya jumlah node. Rangkuman perbandingan kedua strategi disajikan pada Tabel 4.

Tabel 4. Perbandingan Nilai Round Trip Time (RTT) Kedua Strategi.

Jumlah Node	Throughput multicast-vanet	Throughput self-learning	Keterangan
30	7,803 ms	4,475 ms	<i>Self-learning</i> lebih baik
60	10,58 ms	6,655 ms	<i>Self-learning</i> lebih baik
90	28,711 ms	20,754 ms	<i>Self-learning</i> lebih baik
120	46,342 ms	35,543 ms	<i>Self-learning</i> lebih baik

Berdasarkan grafik pada Gambar 8 dapat disimpulkan bahwa strategi self-learning memiliki nilai rata-rata RTT yang lebih rendah dibandingkan multicast-vanet pada seluruh kondisi pengujian. Hal ini terjadi karena waktu pengambilan (fetch time) dipengaruhi oleh jumlah paket yang tersimpan di dalam cache; semakin banyak paket data yang tersimpan di dalam cache, semakin kecil nilai waktu pengambilannya. Strategi self-learning memperoleh keunggulan pada aspek ini karena setelah melakukan broadcast paket yang diinginkan untuk pertama kali ke seluruh antarmuka pada FIB, self-

learning menetapkan satu jalur untuk memperoleh data dari paket interest yang diminta konsumen. Penetapan satu jalur ini mengurangi waktu komputasi untuk melakukan broadcast paket interest ke seluruh antarmuka yang tersedia pada FIB, sehingga waktu pengiriman menjadi lebih singkat dibandingkan strategi multicast yang secara terus-menerus melakukan broadcast ke seluruh jalur. Keunggulan self-learning dalam menekan RTT ini menjadikannya lebih sesuai untuk mendukung komunikasi VANET yang menuntut latensi rendah.



Gambar 8. Grafik Hasil Simulasi Perubahan Jumlah *Node* terhadap *Round-Trip Time*

4. KESIMPULAN

Berdasarkan hasil dan analisis data yang dilakukan terkait perbandingan performansi strategi forwarding multicast-vanet dan self-learning pada jaringan VANET-NDN dengan skenario jalur kendaraan peta grid, dapat disimpulkan beberapa hal sebagai berikut. Pada parameter Cache Hit Ratio, strategi self-learning secara umum memperoleh nilai yang lebih tinggi dibandingkan multicast-vanet. Hal ini disebabkan oleh kemampuan self-learning dalam mempelajari jalur menuju cache, sehingga paket interest dapat dipenuhi oleh data yang tersimpan pada cache secara lebih efektif dibandingkan strategi multicast yang murni berbasis broadcast.

Pada parameter Throughput, strategi self-learning menghasilkan nilai yang praktis setara dengan multicast-vane. Hal ini menunjukkan bahwa proses pembelajaran jalur pada self-learning tidak mengurangi kapasitas pengiriman data. Nilai Throughput kedua strategi meningkat seiring bertambahnya jumlah node, karena semakin banyak node berarti semakin banyak tautan dan sumber data yang dapat dimanfaatkan.

Pada parameter Round-Trip Time (RTT), strategi self-learning secara umum menghasilkan nilai yang lebih rendah dibandingkan multicast-vanet. Nilai RTT semakin kecil seiring bertambahnya jumlah node karena bertambahnya jumlah paket yang tersimpan pada cache mempercepat waktu pengambilan data. Keunggulan self-learning pada aspek ini bersumber dari mekanismenya yang menetapkan satu jalur unicast setelah proses penemuan awal, sehingga menekan waktu komputasi dan overhead dibandingkan strategi multicast yang terus-menerus melakukan broadcast.

Secara keseluruhan, hasil penelitian menunjukkan bahwa strategi self-learning forwarding memberikan efisiensi yang lebih baik dibandingkan strategi multicast-vanet pada jaringan VANET-NDN, terutama dalam hal pemanfaatan cache dan waktu pengambilan data, sementara kapasitas pengiriman (Throughput) tetap terjaga setara. Dengan demikian, self-learning forwarding merupakan pilihan strategi yang lebih sesuai untuk mendukung komunikasi pada jaringan VANET berbasis NDN yang menuntut latensi rendah dan pemanfaatan sumber daya yang hemat. Penelitian selanjutnya dapat diarahkan pada pengujian dengan skenario mobilitas dan topologi jalan yang lebih beragam, serta pada evaluasi pengaruh ukuran cache dan pola permintaan konten terhadap performansi kedua strategi.

DAFTAR PUSTAKA

- Afanasyev, A., Moiseenko, I., & Zhang, L. (2012). *ndnSIM: NDN simulator for NS-3* (NDN Technical Report NDN-0005). NDN Project.
- Ahed, K., Benamar, M., Lahcen, A. A., & Ouazzani, R. E. (2022). Forwarding strategies in vehicular named data networks: A survey. *Journal of King Saud University - Computer and Information Sciences*, 34(5), 1819–1835. <https://doi.org/10.1016/j.jksuci.2020.06.014>.
- Al-Qutwani, M., Wang, X., & Yi, B. (2021). Request/advertise-based content forwarding in vehicular named data networking. *IEEE Access*, 9, 226–236. <https://doi.org/10.1109/ACCESS.2020.3046846>.
- Alkwai, L., Belghith, A., Gazdar, A., & Al-Ahmadi, S. (2022). Comparative analysis of producer mobility management approaches in named data networking. *Applied Sciences*, 12(24), Article 12581. <https://doi.org/10.3390/app122412581>.
- Amadeo, M., Campolo, C., & Molinaro, A. (2012). CRoWN: Content-centric networking in vehicular ad hoc networks. *IEEE Communications Letters*, 16(9), 1380–1383. <https://doi.org/10.1109/LCOMM.2012.072012.120282>.
- Amadeo, M., Campolo, C., & Molinaro, A. (2016). Information-centric networking for connected vehicles: A survey and future perspectives. *IEEE Communications Magazine*, 54(2), 98–104. <https://doi.org/10.1109/MCOM.2016.7402268>.
- Amadeo, M., Ruggeri, G., Campolo, C., & Molinaro, A. (2021). Diversity-improved caching of popular transient contents in vehicular named data networking. *Computer Networks*, 184, Article 107625. <https://doi.org/10.1016/j.comnet.2020.107625>.
- Araujo, G., Peixoto, M., & Sampaio, L. (2022). NDN4IVC: A framework for simulations of realistic VANETs applications through NDN. *Proceedings of the 9th ACM Conference on Information-Centric Networking (ICN)*, 162–164. <https://doi.org/10.1145/3517212.3559483>.
- Chen, C., Jiang, J., Fu, R., Chen, L., Li, C., & Wan, S. (2022). An intelligent caching strategy considering time-space characteristics in vehicular named data networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(10), 19655–19667. <https://doi.org/10.1109/TITS.2021.3128012>.
- Chen, M., Mau, D. O., Zhang, Y., Taleb, T., & Leung, V. C. M. (2014). VENDNET: Vehicular Named Data Network. *Vehicular Communications*, 1(4), 208–213. <https://doi.org/10.1016/j.vehcom.2014.09.002>.
- Grassi, G., Pesavento, D., Pau, G., Vuyyuru, R., Wakikawa, R., & Zhang, L. (2014). VANET via named data networking. *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 410–415. <https://doi.org/10.1109/INFCOMW.2014.6849267>.
- Grassi, G., Pesavento, D., Pau, G., Zhang, L., & Fdida, S. (2015). Navigo: Interest forwarding by geolocations in vehicular named data networking. *2015 IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 1–10. <https://doi.org/10.1109/WoWMoM.2015.7158165>.

- Heidari, A., Jabraeil Jamali, M. A., & Jafari Navimipour, N. (2025). Fuzzy logic multicriteria decision-making for broadcast storm resolution in vehicular ad hoc networks. *International Journal of Communication Systems*, 38(5), Article e6034. <https://doi.org/10.1002/dac.6034>.
- Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., & Braynard, R. L. (2009). Networking named content. *Proceedings of the 5th ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT)*, 1–12. <https://doi.org/10.1145/1658939.1658941>.
- Khan, S. A., & Lim, H. (2024). Real-time vehicle tracking-based data forwarding using RLS in vehicular named data networking. *IEEE Transactions on Intelligent Transportation Systems*, 25(10), 14054–14069. <https://doi.org/10.1109/TITS.2024.3395184>.
- Khan, S. A., Lim, H., & Chaudhry, S. A. (2025). Adaptive data forwarding scheme for diverse infotainment services in vehicular named data networking. *IEEE Transactions on Vehicular Technology*, 74(5), 8012–8022. <https://doi.org/10.1109/TVT.2024.3524122>.
- Khelifi, H., Luo, S., Nour, B., Moun gla, H., Faheem, Y., Hussain, R., & Ksentini, A. (2020). Named data networking in vehicular ad hoc networks: State-of-the-art and challenges. *IEEE Communications Surveys & Tutorials*, 22(1), 320–351. <https://doi.org/10.1109/COMST.2019.2894816>.
- Kuai, M., Hong, X., & Flores, R. R. (2014). Evaluating interest broadcast in vehicular named data networking. *2014 Third GENI Research and Educational Experiment Workshop (GREE)*, 77–78. <https://doi.org/10.1109/GREE.2014.23>.
- Liang, T., Pan, J., Rahman, M. A., Shi, J., Pesavento, D., Afanasyev, A., & Zhang, B. (2020). Enabling named data networking forwarder to work out-of-the-box at edge networks. *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 1–6. <https://doi.org/10.1109/ICCWorkshops49005.2020.9145304>.
- Liao, C., Liu, X., & Zhou, H. (2024). Environment-adaptive dynamic caching for vehicular named data networks in dynamic network environments. *IEEE Transactions on Vehicular Technology*, 73(4), 5861–5871. <https://doi.org/10.1109/TVT.2023.3335990>.
- Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., & Wießner, E. (2018). Microscopic traffic simulation using SUMO. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2575–2582. <https://doi.org/10.1109/ITSC.2018.8569938>.
- Mastorakis, S., Afanasyev, A., & Zhang, L. (2017). On the evolution of ndnSIM: An open-source simulator for NDN experimentation. *ACM SIGCOMM Computer Communication Review*, 47(3), 19–33. <https://doi.org/10.1145/3138808.3138812>.
- Melati, S. R., Yovita, L. V., & Mayasari, R. (2021). Caching performance of named data networking with NDNS. *2021 International Conference on Information Networking (ICOIN)*, 261–266. <https://doi.org/10.1109/ICOIN50884.2021.9333907>.
- Ngo, M., Ohzahata, S., Yamamoto, R., & Kato, T. (2023). A visual-identification based forwarding strategy for vehicular named data networking. *IEICE Transactions on Information and Systems*, E106.D(2), 204–217. <https://doi.org/10.1587/transinf.2022EDP7084>.
- Rahman, M. A., & Zhang, B. (2021). On data-centric forwarding in mobile ad-hoc networks: Baseline design and simulation analysis. *2021 International Conference on Computer Communications and Networks (ICCCN)*, 1–9. <https://doi.org/10.1109/ICCCN52240.2021.9522163>.
- Saxena, D., Raychoudhury, V., Suri, N., Becker, C., & Cao, J. (2016). Named data networking: A survey. *Computer Science Review*, 19, 15–55. <https://doi.org/10.1016/j.cosrev.2016.01.001>.
- Shi, J., Newberry, E., & Zhang, B. (2017). On broadcast-based self-learning in named data networking. *2017 IFIP Networking Conference (IFIP Networking) and Workshops*, 1–9. <https://doi.org/10.23919/IFIPNetworking.2017.8264832>.

- Wang, F., & Hou, R. (2024). Delayed broadcast-based data forwarding strategy for vehicular named data networks. *2024 International Conference on Computing, Networking and Communications (ICNC)*, 133–138. <https://doi.org/10.1109/ICNC59896.2024.10556330>.
- Wang, J., Luo, J., Ran, Y., Yang, J., Liu, K., & Guo, S. (2023). Towards predictive forwarding strategy in vehicular named data networking. *IEEE Transactions on Vehicular Technology*, 72(3), 3751–3763. <https://doi.org/10.1109/TVT.2022.3209539>.
- Wang, L., Afanasyev, A., Kuntz, R., Vuyyuru, R., Wakikawa, R., & Zhang, L. (2012). Rapid traffic information dissemination using named data. *Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design (NoM)*, 7–12. <https://doi.org/10.1145/2248361.2248365>.
- Xylomenos, G., Ververidis, C. N., Siris, V. A., Fotiou, N., Tsilopoulos, C., Vasilakos, X., Katsaros, K. V., & Polyzos, G. C. (2014). A survey of information-centric networking research. *IEEE Communications Surveys & Tutorials*, 16(2), 1024–1049. <https://doi.org/10.1109/SURV.2013.070813.00063>.
- Yi, C., Afanasyev, A., Moiseenko, I., Wang, L., Zhang, B., & Zhang, L. (2013). A case for stateful forwarding plane. *Computer Communications*, 36(7), 779–791. <https://doi.org/10.1016/j.comcom.2013.01.005>.
- Yi, C., Afanasyev, A., Wang, L., Zhang, B., & Zhang, L. (2012). Adaptive forwarding in named data networking. *ACM SIGCOMM Computer Communication Review*, 42(3), 62–67. <https://doi.org/10.1145/2317307.2317319>.
- Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K. C., Crowley, P., Papadopoulos, C., Wang, L., & Zhang, B. (2014). Named data networking. *ACM SIGCOMM Computer Communication Review*, 44(3), 66–73. <https://doi.org/10.1145/2656877.2656887>.