

OPTIMASI PROTOKOL ROUTING DSDV MENGGUNAKAN ALGORITMA TIME PREDICTION PADA JARINGAN MANET

(OPTIMIZATION OF DSDV ROUTING PROTOCOL USING TIME PREDICTION ALGORITHM IN MANET NETWORK)

Baiq Safitri Rabiah Adawiyah¹⁾, Andy Hidayat Jatmika²⁾, dan Raphael Bianco Huwae³⁾

^{1,2,3)}Program Studi Teknik Informatika, Fakultas Teknik Universitas Mataram

Jl. Majapahit No.62, Gomong, Kec. Selaparang, Kota Mataram, Nusa Tenggara Bar. 83125

e-mail: baigsafitrirabiahadawiyah@gmail.com¹⁾, andy@staff.unram.ac.id²⁾,

raphael.bianco.huawe@unram.ac.id³⁾

ABSTRAK

Protokol Destination Sequenced Distance Vector (DSDV) pada jaringan Mobile Ad Hoc Network (MANET) menggunakan metode pembaruan tabel routing secara terus-menerus dan penyebaran pesan secara broadcast. Pendekatan ini menyebabkan peningkatan lalu lintas jaringan, interferensi sinyal, serta ketidakstabilan rute pada lingkungan dengan mobilitas tinggi. Akibatnya, komunikasi menjadi tidak efisien, sering kehilangan paket data, dan performa jaringan menurun, terutama dalam kondisi dengan kepadatan node yang tinggi. Penelitian ini bertujuan mengoptimalkan kinerja DSDV dengan mengembangkan algoritma prediksi waktu tinggal node, yang dinamakan TimePrediction-DSDV. Algoritma ini memodifikasi pesan Hello dengan menambahkan informasi posisi dan kecepatan node tetangga untuk memprediksi stabilitas rute dan mengurangi dampak interferensi sinyal. Penelitian ini bertujuan mengoptimalkan kinerja protokol DSDV dengan menerapkan algoritma prediksi waktu tinggal node, yang disebut TimePrediction-DSDV. Algoritma ini menggunakan modifikasi pesan Hello untuk menyertakan informasi posisi dan kecepatan node tetangga guna memprediksi stabilitas rute dan mengurangi dampak interferensi. Metodologi melibatkan simulasi menggunakan Network Simulator 2.35 pada area 1500x1500 meter dengan skenario kepadatan node 25, 50, dan 100 serta kecepatan 30, 60, dan 90 m/s. Parameter kinerja yang diuji meliputi throughput, end-to-end delay, dan Packet Delivery Ratio (PDR). Hasil penelitian menunjukkan bahwa TimePrediction-DSDV mampu meningkatkan throughput hingga 15%, menurunkan delay hingga 22%, dan meningkatkan PDR hingga 12.5% dibandingkan DSDV standar. Algoritma ini terbukti efektif dalam meningkatkan performa jaringan meskipun kepadatan dan kecepatan node bertambah.

Kata kunci: DSDV, Prediksi Waktu, MANET, Pergerakan Node, Routing.

ABSTRACT

The Destination Sequenced Distance Vector (DSDV) protocol in Mobile Ad Hoc Networks (MANETs) employs a method of continuously updating routing tables and broadcasting messages. This approach increases network traffic, signal interference, and route instability in high-mobility environments. Consequently, communication becomes inefficient, with frequent packet losses and degraded network performance, especially in high node density conditions. This study aims to optimize the performance of DSDV by developing a node residual time prediction algorithm, named TimePrediction-DSDV. The algorithm modifies Hello messages by incorporating positional and velocity information of neighboring nodes to predict route stability and mitigate the impact of signal interference. The methodology involves simulations conducted using Network Simulator 2.35 on a 1500x1500-meter area with scenarios of 25, 50, and 100 node densities and node speeds of 30, 60, and 90 m/s. The performance parameters evaluated include throughput, end-to-end delay, and Packet Delivery Ratio (PDR). The results show that TimePrediction-DSDV can improve throughput by up to 15%, reduce delay by up to 22%, and increase PDR by up to 12.5% compared to standard DSDV. This algorithm proves to be effective in enhancing network performance, even as node density and speed increase.

Keywords: DSDV, TimePrediction, MANET, Node Mobility, Routing.

I. PENDAHULUAN

Protokol routing Destination Sequenced Distance Vector (DSDV) adalah salah satu protokol routing yang digunakan dalam jaringan Mobile Ad Hoc Network (MANET)[1]. Protokol ini termasuk

dalam kategori protokol routing proaktif yang secara berkala memperbarui informasi routing ke seluruh node dalam jaringan[2]. Setiap node dalam jaringan secara terus-menerus berbagi informasi routing dengan node tetangganya, sehingga dapat langsung merespons permintaan untuk rute tanpa perlu mencari rute secara on-demand[3]. Dalam jaringan MANET, node-node

bergerak secara bebas dan tidak teratur sehingga menyebabkan topologi jaringan selalu berubah. Dalam lingkungan jaringan yang dinamis dan tanpa infrastruktur seperti MANET, pemilihan rute yang optimal adalah hal yang sangat penting[4].

Protokol routing DSDV menggunakan prinsip tabel routing berbasis *hop count* ketika memilih rute[5]. Proses pengiriman pesan *Route Request* (RREQ) pada protokol routing DSDV adalah mekanisme yang digunakan untuk menemukan rute ke tujuan dalam jaringan MANET. Node yang mengirimkan RREQ (node sumber) akan menyebarkan paket ini ke semua node tetangganya. Setiap node tetangga yang menerima RREQ akan memeriksa tabel routingnya untuk melihat apakah mereka memiliki rute langsung ke tujuan yang diminta. Jika node tetangga tidak memiliki rute langsung ke tujuan, node tetangga akan meneruskan RREQ ke tetangga lainnya. RREQ akan terus menyebar melalui jaringan sampai mencapai node yang memiliki rute langsung atau menemukan node yang lebih dekat ke tujuan[6].

Dalam proses penyebaran RREQ, DSDV tidak melakukan mekanisme untuk memprediksi atau mempertimbangkan mobilitas node, akibatnya rute yang ditemukan dapat menjadi tidak valid jika node-node yang berada di sepanjang jalur rute tersebut bergerak menjauh dan keluar dari jangkauan komunikasi[3]. Selain itu, pendekatan *broadcast* yang digunakan DSDV untuk menyebarkan RREQ dapat menyebabkan peningkatan lalu lintas jaringan dan interferensi antar node, sehingga berpotensi menurunkan kinerja jaringan, terutama dalam lingkungan dengan kepadatan node yang tinggi[7].

Terdapat beberapa penelitian yang telah dilakukan menggunakan protokol routing DSDV dalam mencari rute dengan berbagai lingkungan simulasi. Pada penelitian [8] dilakukan penelitian menggunakan protokol routing DSDV yang diuji pada lingkungan simulasi dengan skenario jumlah node 10, 20, dan 40. Parameter yang digunakan untuk menguji kinerja jaringan adalah *Packet Delivery Ratio* (PDR), *throughput*, dan *delay*. Hasil pengujian menunjukkan bahwa nilai PDR pada protokol routing DSDV memberikan hasil yang kurang baik yaitu dibawah 55%. Pada penelitian tersebut tidak dilakukan pertimbangan perhitungan prediksi pergerakan node. Pada penelitian [9] penulis melakukan penelitian menggunakan protokol routing proaktif DSDV dengan mengambil skenario studi kasus gempa Palu 2018 yang diuji menggunakan jumlah node sebanyak 60, 80, 100, dan 150. Parameter uji kinerja jaringan diukur menggunakan *throughput*, *delay*, dan *packet loss*. Hasil penelitian menunjukkan bahwa semakin besar kepadatan node maka nilai *throughput* yang diberikan

semakin turun. Hal ini kemungkinan terjadi akibat interferensi sinyal antar node pada saat pencarian rute. Pada penelitian tersebut tidak dilakukan perhitungan prediksi pergerakan node.

Tujuan dari penelitian ini adalah mengoptimalkan kinerja protokol routing DSDV dengan cara memilih node tetangga dengan mempertimbangkan arah mobilitas node dalam proses penemuan rute. Metode yang digunakan adalah dengan memodifikasi pesan Hello untuk menyertakan informasi posisi dan kecepatan node tetangga. Informasi ini kemudian akan digunakan untuk memprediksi waktu tinggal node di dalam jangkauan komunikasi, sehingga rute yang dipilih memiliki probabilitas putus yang lebih rendah. Selain itu, dengan membatasi penyebaran pesan pencarian rute hanya pada node yang relevan, diharapkan dapat mengurangi interferensi dan meningkatkan efisiensi jaringan.

Simulasi dan modifikasi protokol routing menggunakan simulator NS-2.35 yang dijalankan pada sistem operasi Linux Ubuntu 14.04 pada VirtualBox. Dalam penelitian ini, protokol routing DSDV standar dijadikan sebagai titik acuan untuk membandingkan kinerja dengan versi DSDV yang telah dimodifikasi. Parameter kinerja yang diukur meliputi *throughput*, *end-to-end delay*, dan *Packet Delivery Ratio* (PDR).

II. STUDI PUSTAKA

Pada penelitian [10], penulis mengembangkan model prediksi waktu berbasis mobilitas node untuk meningkatkan efisiensi protokol *Optimized Link State Routing* (OLSR) dalam jaringan *Ad Hoc Mobile* (MANET). Penelitian ini memperhitungkan posisi dan kecepatan node untuk memprediksi waktu yang diperlukan oleh node tetangga untuk keluar dari jangkauan komunikasi. Modifikasi dilakukan dengan mengubah interval pesan HELLO secara dinamis, yang sebelumnya statis pada OLSR standar. Pengujian dilakukan dalam lingkungan simulasi menggunakan 30 node dengan model mobilitas *Random Waypoint* dalam jangkauan nirkabel 350 meter. Parameter yang dievaluasi meliputi *packet loss* dan *delay*. Hasil penelitian menunjukkan bahwa modifikasi pada OLSR mampu mengurangi paket yang hilang dan memperbaiki total *delay* dibandingkan dengan versi standar, menunjukkan peningkatan kualitas layanan (QoS) dalam jaringan dengan mobilitas tinggi.

Pada penelitian [8], dilakukan analisis performa protokol routing proaktif dan reaktif pada MANET dengan menggunakan simulator

OMNeT++. Penelitian ini menguji protokol DSDV (proaktif) dan AODV (reaktif) dengan variasi kepadatan node 10, 20, dan 40 node. Parameter evaluasi meliputi *Packet Delivery Ratio* (PDR), *throughput*, dan *delay*. Hasil analisis menunjukkan bahwa pergerakan node berdampak signifikan terhadap kinerja jaringan, khususnya pada nilai *Packet Delivery Ratio* (PDR). Protokol AODV menunjukkan keunggulan dalam hal PDR dan *delay*, sedangkan protokol DSDV menunjukkan kestabilan *throughput* yang lebih baik pada kepadatan node yang rendah.

Pada penelitian [11], penulis melakukan analisis pengaruh perubahan jumlah node terhadap kinerja protokol routing DSDV dengan menggunakan beberapa variasi jumlah node sebanyak 20, 40, 60 dan 80 dengan kecepatan node 20 m/s. Parameter pengujian mencakup *throughput* dan *delay*. Hasil pengujian menunjukkan bahwa mobilitas node berdampak signifikan terhadap penurunan *throughput* seiring bertambahnya jumlah node, dari 291 kbps menjadi 64,3 kbps. Namun, nilai *delay* mengalami perbaikan, menurun dari 454,7 ms menjadi 322,4 ms, menunjukkan sifat proaktif DSDV yang efisien dalam memperbarui tabel routing.

Pada penelitian [12], penulis mencoba menganalisis pengaruh pola mobilitas node terhadap kinerja protokol AODV pada MANET menggunakan emulator Mininet-WiFi. Penelitian ini menguji dua model mobilitas, yaitu *Gauss Markov* dan *Random Direction*, yang diterapkan pada skenario dengan jumlah node sebanyak 10 node. Parameter evaluasi meliputi *throughput*, *latency average*, *jitter*, dan *packet loss*. Hasil penelitian menunjukkan bahwa model *Gauss Markov* mampu menghasilkan *throughput* yang lebih tinggi dan *packet loss* yang lebih rendah dibandingkan *Random Direction*. Hal ini menunjukkan bahwa model *Gauss Markov* lebih efektif dalam mengatasi tantangan perubahan topologi jaringan yang bersifat dinamis, sehingga meningkatkan stabilitas kinerja protokol AODV.

Penelitian [13] menganalisis dampak pola pergerakan node terhadap konsumsi energi protokol *Optimized Link State Routing* (OLSR) pada MANET. Model mobilitas yang diuji meliputi *Random Waypoint*, *Random Walk*, dan *Random Direction*, dengan variasi jumlah node sebanyak 20, 40, 60, 80, dan 100. Parameter pengujian yang digunakan adalah konsumsi energi total, energi sisa, dan rata-rata konsumsi energi. Hasil menunjukkan bahwa mobilitas *Random Direction*

memiliki konsumsi energi paling rendah karena pola gerakannya yang lebih efisien, sedangkan *Random Waypoint* menunjukkan konsumsi energi tertinggi akibat pola pergerakan yang kurang optimal.

Sebagian besar penelitian sebelumnya memberikan landasan penting terkait kelemahan protokol routing seperti DSDV, AODV, dan OLSR dalam menghadapi mobilitas tinggi dan kepadatan node yang bervariasi. Persamaan dengan penelitian ini terletak pada fokus evaluasi performa protokol routing berdasarkan parameter seperti *throughput*, *end-to-end delay*, dan *Packet Delivery Ratio* (PDR), yang menjadi indikator utama efisiensi jaringan. Namun, pendekatan berbasis prediksi mobilitas node untuk meningkatkan stabilitas rute masih jarang dilakukan. Penelitian ini menawarkan solusi dengan memodifikasi DSDV menggunakan algoritma prediksi waktu tinggal node, yang mengintegrasikan informasi posisi dan kecepatan untuk meningkatkan efisiensi jaringan.

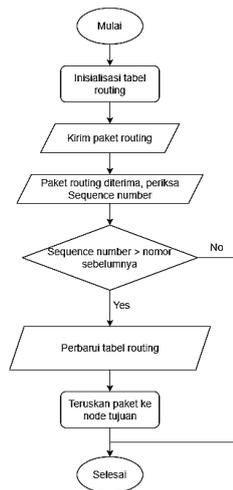
III. METODE PENELITIAN

Penelitian ini dilakukan melalui serangkaian tahapan yang bertujuan untuk menganalisis kinerja protokol *Destination-Sequenced Distance Vector* (DSDV), mekanisme prediksi mobilitas node tetangga, efektivitas pesan Hello, serta performa jaringan berdasarkan parameter yang telah ditentukan.

A. Mekanisme kerja protokol DSDV.

Protokol *Destination Sequenced Distance Vector* (DSDV) adalah protokol routing proaktif yang secara berkala memperbarui tabel routing di setiap node dalam jaringan *ad-hoc*. Setiap tabel berisi informasi rute ke semua tujuan, termasuk jumlah hop dan nomor urut (*sequence number*), yang membantu mencegah loop routing dan memastikan konsistensi informasi[1].

Proses alur kerja protokol DSDV dapat dilihat pada Gambar 1. *Flowchart* di bawah ini, yang menggambarkan tahapan utama dalam pembaruan tabel routing dan pengiriman paket data di protokol DSDV

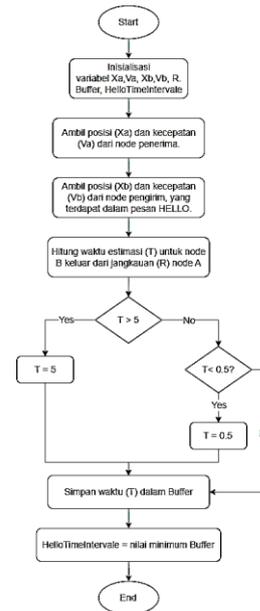


Gambar 1. Flowchart Protokol DSDV

Berdasarkan Gambar 1, *alur kerja* protokol *Destination Sequenced Distance Vector (DSDV)* dapat dijelaskan sebagai berikut:

1. Proses dimulai dengan inisialisasi tabel routing di setiap node. Tabel routing berisi informasi tentang rute ke semua node dalam jaringan, termasuk nomor urut (*sequence number*) dan jumlah hop sebagai metrik.
2. Setelah inisialisasi, setiap node mengirimkan paket routing yang berisi informasi tabel routing kepada node tetangganya.
3. Ketika sebuah node menerima paket routing, nomor urut pada paket diperiksa untuk menentukan validitasnya:
 - Jika nomor urut lebih besar daripada nomor yang tercatat sebelumnya, paket dianggap valid, dan tabel routing diperbarui berdasarkan informasi terbaru dari paket tersebut.
 - Sebaliknya, jika nomor urut lebih kecil atau sama dengan nomor yang tercatat, paket dianggap usang atau duplikat sehingga diabaikan.
4. Setelah tabel routing diperbarui, node menggunakan informasi dalam tabel tersebut untuk mengirimkan paket data ke node tujuan melalui rute terbaik.
5. Proses ini terus berulang secara berkala atau ketika terjadi perubahan topologi jaringan, untuk memastikan bahwa setiap node memiliki informasi rute terbaru dan dapat mengirimkan data dengan efisien.

B. Algoritma prediksi mobilitas node



Gambar 2. Flowchart Mobilitas Node

Flowchart pada Gambar 2 di atas menjelaskan mekanisme perhitungan interval waktu node tetangga untuk keluar dari jangkauan node utama dalam jaringan MANET. Proses dimulai dengan inisialisasi beberapa variabel, yaitu posisi dan kecepatan node penerima (X_a dan V_a), posisi dan kecepatan node pengirim (X_b dan V_b) yang diperoleh melalui pesan HELLO, radius jangkauan komunikasi (R), *buffer* untuk menyimpan nilai waktu estimasi, serta *HelloTimeInterval* sebagai interval waktu pengiriman pesan HELLO. Setelah inisialisasi, data posisi dan kecepatan dari kedua node dihimpun untuk digunakan dalam perhitungan waktu estimasi (T), yang bertujuan menentukan kapan node pengirim keluar dari jangkauan node penerima.

Nilai waktu estimasi yang dihitung selanjutnya disesuaikan dengan batasan tertentu. Jika nilai T lebih besar dari 5, maka waktu estimasi dibatasi hingga maksimum 5 untuk menghindari interval waktu yang terlalu lama. Sebaliknya, jika nilai T kurang dari 0.5, maka waktu estimasi diatur ke nilai minimum 0.5 untuk memastikan pengiriman pesan tetap berlangsung secara reguler. Nilai waktu estimasi ini kemudian disimpan dalam *buffer*, dan nilai minimum dari *buffer* digunakan untuk menentukan *HelloTimeInterval*. Proses ini memastikan bahwa interval waktu pengiriman pesan HELLO bersifat adaptif terhadap kondisi mobilitas node, sehingga *overhead* komunikasi

dapat diminimalkan dan stabilitas jaringan MANET dapat ditingkatkan.

C. Pesan Hello

Pesan HELLO dalam jaringan *Mobile Ad Hoc Network* (MANET) berfungsi sebagai mekanisme bagi setiap node untuk secara berkala menginformasikan keberadaannya kepada node tetangga. Tujuan utama dari pesan ini adalah untuk mendeteksi dan memelihara informasi konektivitas dengan node tetangga, sehingga membantu dalam pembaruan tabel routing dan memastikan rute yang optimal dalam jaringan[4]. Dalam protokol *Destination Sequenced Distance Vector* (DSDV), setiap node secara berkala mengirimkan pesan HELLO untuk menunjukkan bahwa node tersebut masih aktif dan dapat dijangkau oleh node tetangga. Pesan HELLO ini membantu dalam pemeliharaan tabel routing dengan memastikan informasi tentang ketersediaan dan status link dengan node tetangga selalu diperbarui[1].

Pemilihan interval pengiriman pesan HELLO sangat berpengaruh pada kinerja jaringan MANET. Interval yang terlalu pendek dapat meningkatkan overhead jaringan, sedangkan interval yang terlalu panjang dapat menyebabkan keterlambatan dalam mendeteksi perubahan topologi. Oleh karena itu, penentuan nilai interval yang optimal menjadi penting untuk menjaga keseimbangan antara overhead dan responsivitas jaringan[4].

Struktur pesan HELLO, seperti yang ditunjukkan pada Tabel 1, mencakup elemen-elemen penting yang dirancang untuk mendukung fungsi tersebut, mulai dari informasi konektivitas hingga detail posisi dan kecepatan node.

Tabel 1. Struktur Pesan Hello

Reserved		Htime	Willingness
Link Code	Reserved	Link Message Size	
Neighbor Interface Address			
Neighbor Interface Address			
Xpos	yPos		
zPos	Vx		
Vy	Vz		

Tabel 1 di atas menunjukkan struktur pesan HELLO yang digunakan dalam jaringan MANET. Pesan ini terdiri dari beberapa elemen penting yang mendukung fungsinya sebagai mekanisme untuk memelihara informasi konektivitas antar node. Kolom *Reserved* digunakan untuk menjaga kesesuaian format data dalam protokol. *Htime*

merupakan waktu estimasi yang menunjukkan durasi validitas pesan HELLO, sedangkan *Willingness* mencerminkan tingkat keinginan sebuah node untuk bertindak sebagai *relay* dalam proses routing. *Elemen Link Code* dan *Link Message Size* memberikan informasi spesifik terkait tipe koneksi dan ukuran pesan, yang membantu dalam pengelolaan informasi topologi.

Selain itu, kolom *Neighbor Interface Address* digunakan untuk mencatat alamat antarmuka dari node tetangga yang berada dalam jangkauan. Elemen tambahan seperti *xPos*, *yPos*, dan *zPos* menyediakan informasi tentang posisi node, sedangkan *Vx*, *Vy*, dan *Vz* memberikan data kecepatan node. Informasi ini sangat penting dalam jaringan MANET yang memiliki topologi dinamis, karena mendukung pembaruan tabel routing secara real-time untuk menjaga efisiensi dan stabilitas jaringan.

D. Parameter kinerja jaringan

Dalam evaluasi kinerja protokol routing pada jaringan, beberapa parameter utama digunakan untuk mengukur efektivitas dan keandalan protokol dalam mengelola transmisi data. Parameter-parameter yang umum digunakan antara lain *throughput*, *end-to-end delay*, dan *Packet Delivery Ratio* (PDR), yang masing-masing memberikan wawasan mengenai kecepatan transfer data, latensi, dan tingkat keberhasilan pengiriman paket dalam jaringan [4]. Penjelasan mengenai masing-masing parameter, beserta rumus perhitungannya, dapat dilihat di bawah ini:

1. Throughput

Throughput merupakan salah satu parameter kinerja utama dalam jaringan MANET yang mengukur jumlah data yang berhasil dikirimkan dari sumber ke tujuan dalam satuan waktu tertentu. Parameter ini digunakan untuk mengevaluasi efisiensi protokol routing dalam mentransmisikan data secara efektif. Dalam jaringan MANET, *throughput* dipengaruhi oleh berbagai faktor, termasuk kepadatan node, kapasitas saluran, dan kecepatan pergerakan node. Nilai *throughput* yang tinggi menunjukkan bahwa jaringan mampu mentransfer data dengan baik, yang menjadi indikator penting kinerja protokol routing. Namun, peningkatan *throughput* sering kali diikuti oleh kompromi terhadap parameter lain, seperti latensi atau keandalan jaringan. Oleh karena itu, pengelolaan *throughput* menjadi tantangan

tersendiri dalam lingkungan jaringan yang dinamis seperti MANET [2]. Rumus perhitungannya ditunjukkan pada Persamaan (1):

$$\text{Throughput} = \frac{\text{Ukuran data diterima (bit)}}{\text{Waktu transmisi (detik)}} \quad (1)$$

2. End to end delay

End-to-end delay mengukur waktu rata-rata yang dibutuhkan oleh paket data untuk berpindah dari sumber ke tujuan. Parameter ini mencakup berbagai komponen *delay*, seperti *delay* propagasi, *delay* antrian, dan *delay* pemrosesan. *End-to-end delay* sangat penting untuk menilai latensi dalam jaringan, terutama bagi aplikasi *real-time* seperti komunikasi suara atau video, yang sangat sensitif terhadap peningkatan *delay*. Dalam kondisi jaringan yang padat atau dengan mobilitas node yang tinggi, nilai *delay* cenderung meningkat, yang dapat berdampak negatif pada kinerja protokol routing. Oleh karena itu, pengurangan *delay* menjadi salah satu prioritas untuk meningkatkan efisiensi jaringan [4]. Rumus untuk menghitung *end-to-end delay* diberikan pada Persamaan (2):

$$\text{Delay} = \frac{\sum(\text{waktu terima} - \text{waktu kirim})}{\text{Jumlah paket yang diterima}} \quad (2)$$

3. Packet Delivery Ratio (PDR)

Packet Delivery Ratio (PDR) adalah parameter yang mengukur persentase paket data yang berhasil diterima oleh tujuan dibandingkan dengan jumlah total paket yang dikirim oleh sumber. Parameter ini digunakan untuk mengevaluasi keandalan protokol routing dalam mengirimkan data tanpa kehilangan paket. Nilai PDR yang tinggi menunjukkan bahwa protokol routing mampu menjaga stabilitas dan efisiensi dalam memilih jalur pengiriman data, bahkan dalam kondisi jaringan yang penuh dengan gangguan atau mobilitas tinggi. PDR menjadi salah satu parameter yang paling sering digunakan untuk menilai performa protokol routing dalam lingkungan MANET yang dinamis [1]. Persamaan (3) berikut digunakan untuk menghitung PDR:

$$\text{PDR} = \frac{\text{Jumlah paket diterima}}{\text{Paket pengiriman data}} \times 100\% \quad (3)$$

IV. HASIL DAN PEMBAHASAN

A. Parameter Skenario Simulasi

Simulasi dilakukan menggunakan Network Simulator (NS-2) versi 2.35 yang berjalan pada sistem operasi Linux Ubuntu 14.04 melalui VirtualBox. Protokol routing DSDV dimodifikasi dengan mengubah modul bawaan NS-2 menggunakan bahasa pemrograman C++. Script

simulasi ditulis dalam *Tool Command Language* (TCL). Hasil simulasi berupa *file trace* berekstensi .tr diproses menggunakan bahasa pemrograman AWK untuk menghitung *throughput*, *end-to-end delay*, dan *Packet Delivery Ratio* (PDR). Simulasi mencakup kepadatan node 25, 50, dan 100 node serta kecepatan 30, 60, dan 90 m/s di area seluas 1500x1500 meter persegi. Model mobilitas *Random Waypoint* digunakan dengan pola trafik CBR dan antena *omni-directional*. Durasi simulasi 300 detik untuk membandingkan kinerja protokol DSDV standar dan DSDV dengan modifikasi berbasis algoritma *TimePrediction*.

Parameter utama yang digunakan dalam simulasi mencakup kepadatan node, kecepatan, dan konfigurasi jaringan lainnya. Detail parameter simulasi dapat dilihat pada Tabel 2 di bawah ini.

Tabel 2. Parameter Skenario Simulasi

Versi Network Simulator	NS-2.35
Jumlah kepadatan node	25, 50, 100 node
Kecepatan node	30, 60, 90 m/s
Luas area simulasi	1500x1500 m ²
Protokol routing	DSDV
Model Mobilitas	Random Waypoint
Waktu simulasi	300 detik
Model antena	Omni-directional
MAC protocol	IEEE 802.11
Pola trafik	CBR

Sebagaimana terlihat pada Tabel 2 di atas, simulasi mencakup area seluas 1500x1500 meter persegi dengan tiga tingkat kepadatan node: 25, 50, dan 100 node. Variasi kecepatan node (30, 60, dan 90 m/s) digunakan untuk mencerminkan kondisi jaringan dinamis. Penggunaan pola trafik CBR dan antena *omni-directional* memungkinkan evaluasi performa protokol routing dalam berbagai skenario jaringan MANET.

B. Evaluasi Kinerja Berdasarkan Parameter Jaringan

Pembahasan ini menyajikan hasil evaluasi kinerja protokol routing DSDV dan *TimePrediction-DSDV* berdasarkan tiga parameter utama jaringan, yaitu *throughput*, *delay*, dan *Packet Delivery Ratio* (PDR). Pengujian dilakukan menggunakan simulator NS-2.35. Hasil pengujian mencakup simulasi pada kepadatan node 25, 50, dan 100 dengan kecepatan tetap sebesar 30 m/s, dan hasil pengujian untuk variasi kecepatan pada kepadatan tertentu disajikan dalam bentuk grafik untuk menunjukkan pengaruh kecepatan dan

kepadatan node terhadap performa protokol routing yang diuji.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f throughput.awk dsdv-hast1.tr
Average Throughput[kbps] = 1998
```

Gambar 3. *Throughput* 25 node Kecepatan 30 m/s

Berdasarkan Gambar 3 di atas, menunjukkan hasil uji coba TimePrediction-DSDV dengan skenario pengukuran *throughput* pada kepadatan node sebanyak 25 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 3 terlihat bahwa hasil *throughput* sebesar 1998 kbps. Untuk hasil uji parameter *throughput* dengan kepadatan 25 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 9.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f delay-pdr.awk dsdv-hast1.tr
Packet Delivery Ratio : 97%
End to End Delay : 0.16
```

Gambar 4. PDR dan *Delay* 25 node Kecepatan 30 m/s

Berdasarkan Gambar 4 di atas, menunjukkan hasil simulasi protokol DSDV dengan skenario pengukuran *Packet Delivery Ratio* (PDR) dan *end-to-end delay* pada kepadatan node sebanyak 25 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 4 terlihat bahwa hasil PDR sebesar 97%, dan *end-to-end delay* 0,16 detik. Untuk hasil uji parameter PDR dan *end-to-end delay* dengan kepadatan 25 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 12 dan Gambar 15.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f throughput.awk dsdv-hast1.tr
Average Throughput[kbps] = 1722
```

Gambar 5. *Throughput* 50 node Kecepatan 30 m/s

Berdasarkan Gambar 5 di atas, menunjukkan hasil uji coba TimePrediction-DSDV dengan skenario pengukuran *throughput* pada kepadatan node sebanyak 50 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 5 terlihat bahwa hasil *throughput* sebesar 1722 kbps. Untuk hasil uji parameter *throughput* dengan kepadatan 50 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 10.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f delay-pdr.awk asdv-hast1.tr
Packet Delivery Ratio : 93%
End to End Delay : 0.237
```

Gambar 6. PDR dan *Delay* 50 node Kecepatan 30 m/s

Berdasarkan Gambar 6 di atas, menunjukkan hasil simulasi protokol DSDV dengan skenario pengukuran *Packet Delivery Ratio* (PDR) dan *end-to-end delay* pada kepadatan node sebanyak 50 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 6 terlihat bahwa hasil PDR sebesar 93%, dan *end-to-end delay* 0,237 detik. Untuk hasil uji parameter PDR dan *end-to-end delay* dengan kepadatan 50 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 13 dan Gambar 16.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f throughput.awk dsdv-hast1.tr
Average Throughput[kbps] = 1550
```

Gambar 7. *Throughput* 100 node Kecepatan 30 m/s

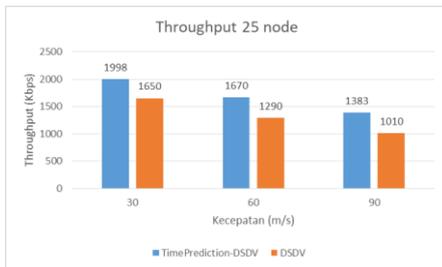
Berdasarkan Gambar 7 di atas, menunjukkan hasil uji coba TimePrediction-DSDV dengan skenario pengukuran *throughput* pada kepadatan node sebanyak 100 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 7 terlihat bahwa hasil *throughput* sebesar 1550 kbps. Untuk hasil uji parameter *throughput* dengan kepadatan 100 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 11.

```
wlya@wlya-VirtualBox:~/TA_wily$ awk -f delay-pdr.awk asdv-hast1.tr
Packet Delivery Ratio : 86%
End to End Delay : 0.303
```

Gambar 8. PDR dan *Delay* 100 node Kecepatan 30 m/s

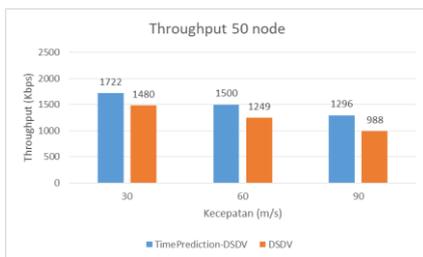
Berdasarkan Gambar 8 di atas, menunjukkan hasil simulasi protokol DSDV dengan skenario pengukuran *Packet Delivery Ratio* (PDR) dan *end-to-end delay* pada kepadatan node sebanyak 100 dan kecepatan 30 m/s. Nilai parameter uji diambil dari *file trace* (.tr) menggunakan *script* AWK. *File* (.tr) merupakan *file* yang dihasilkan dari simulasi yang dijalankan menggunakan NS-2. Pada Gambar 8 terlihat bahwa hasil PDR sebesar 86%, dan *end-to-end delay* 0,303 detik. Untuk hasil uji parameter PDR dan *end-to-end delay* dengan kepadatan 100 node dan variasi kecepatan yang berbeda, hasilnya dapat dilihat pada Gambar 14 dan Gambar 17.

Di bawah ini disajikan grafik-grafik hasil simulasi yang menunjukkan pengaruh kepadatan node 25, 50, dan 100 serta kecepatan 30, 60, dan 90 m/s di area seluas 1500x1500 meter persegi terhadap performa protokol DSDV dan TimePrediction-DSDV berdasarkan parameter *throughput*, *Packet Delivery Ratio* (PDR), dan *end-to-end delay*.



Gambar 9. Grafik *Throughput* 25 node

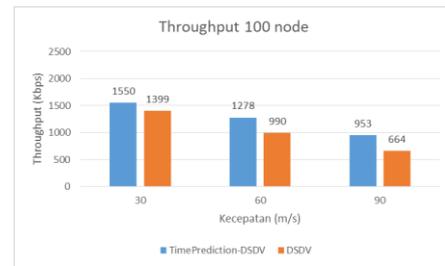
Berdasarkan Gambar 3. Grafik di atas menunjukkan bahwa nilai *throughput* menurun seiring dengan meningkatnya kecepatan node pada kepadatan 25 node. TimePrediction-DSDV mampu meningkatkan *throughput* dibandingkan DSDV standar sebesar 18.00%, 15.00%, dan 12.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Peningkatan *throughput* ini terjadi karena algoritma prediksi pada TimePrediction-DSDV dapat mempertahankan rute yang lebih stabil, meskipun perubahan topologi jaringan semakin dinamis akibat peningkatan kecepatan node.



Gambar 10. Grafik *Throughput* 50 node

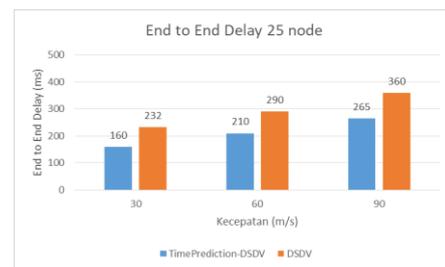
Berdasarkan Gambar 4. Grafik di atas menunjukkan bahwa nilai *throughput* pada kepadatan 50 node menurun lebih signifikan dibandingkan pada 25 node. TimePrediction-DSDV mampu meningkatkan *throughput* dibandingkan DSDV standar sebesar 17.50%, 14.50%, dan 11.50% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Pada kepadatan yang lebih tinggi, potensi interferensi meningkat, tetapi algoritma prediktif TimePrediction-DSDV membantu menjaga

stabilitas rute, sehingga *throughput* tetap lebih baik dibandingkan DSDV standar.



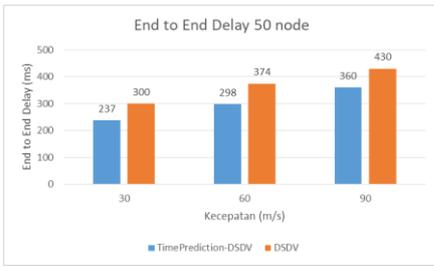
Gambar 11. Grafik *Throughput* 100 node

Berdasarkan Gambar 5. Grafik *Throughput* pada kepadatan 100 node mengalami penurunan paling tajam dibandingkan kepadatan lainnya. TimePrediction-DSDV tetap unggul dengan peningkatan *throughput* sebesar 16.00%, 13.00%, dan 10.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Kepadatan jaringan yang tinggi meningkatkan peluang tabrakan data, tetapi TimePrediction-DSDV mampu mengurangi dampaknya melalui prediksi rute yang lebih baik.



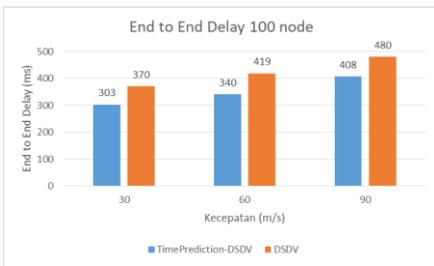
Gambar 12. Grafik *End to End Delay* 25 node

Berdasarkan Gambar 6. Grafik di atas menunjukkan bahwa nilai *end-to-end delay* menurun pada protokol TimePrediction-DSDV dibandingkan dengan DSDV standar pada kepadatan 25 node. Penurunan *delay* tercatat sebesar 25.00%, 22.00%, dan 20.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. TimePrediction-DSDV mampu mengurangi *delay* secara signifikan karena algoritma prediksinya dapat mempertahankan rute yang lebih stabil, sedangkan DSDV standar membutuhkan waktu lebih lama untuk memperbarui tabel routing.



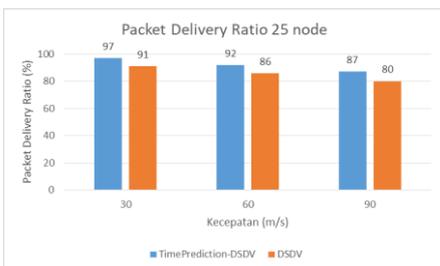
Gambar 13. Grafik *End to End Delay* 50 node

Berdasarkan Gambar 7. Grafik di atas, *delay* meningkat pada kepadatan 50 node, tetapi TimePrediction-DSDV tetap menunjukkan *delay* yang lebih rendah dibandingkan DSDV standar. Penurunan *delay* pada TimePrediction-DSDV tercatat sebesar 24.00%, 20.50%, dan 18.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Peningkatan kepadatan node menyebabkan lebih banyak antrian data di jaringan, namun algoritma prediktif TimePrediction-DSDV membantu meminimalkan dampak ini.



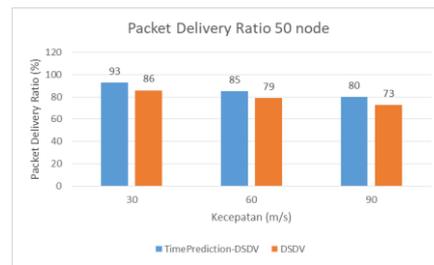
Gambar 14. Grafik *End to End Delay* 100 node

Berdasarkan Gambar 8. Grafik di atas terlihat bahwa *delay* meningkat paling signifikan pada kepadatan 100 node, terutama pada kecepatan tinggi. TimePrediction-DSDV mampu menurunkan *delay* sebesar 22.50%, 19.00%, dan 17.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Kepadatan node yang tinggi meningkatkan peluang terjadinya antrian paket dan waktu pemrosesan, namun algoritma prediktif TimePrediction-DSDV tetap unggul dalam menjaga efisiensi.



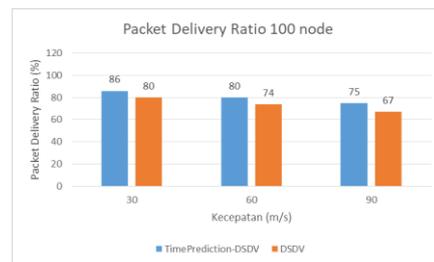
Gambar 15. Grafik *Packet Delivery Ratio* 25 node

Berdasarkan Gambar 9. Grafik di atas menunjukkan bahwa nilai PDR pada TimePrediction-DSDV lebih tinggi dibandingkan dengan DSDV standar pada seluruh kecepatan dengan kepadatan 25 node. Peningkatan PDR tercatat sebesar 15.00%, 12.50%, dan 10.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. TimePrediction-DSDV menunjukkan performa lebih baik karena kemampuannya untuk mempertahankan rute yang lebih stabil, sehingga mengurangi kehilangan paket data akibat perubahan topologi.



Gambar 16. Grafik *Packet Delivery Ratio* 50 node

Berdasarkan Gambar 10. Grafik di atas, PDR menurun pada kepadatan 50 node dibandingkan dengan 25 node, tetapi TimePrediction-DSDV tetap unggul dibandingkan DSDV standar. Peningkatan PDR tercatat sebesar 14.50%, 11.50%, dan 9.00% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Kepadatan jaringan yang lebih tinggi meningkatkan peluang terjadinya collision, namun algoritma prediksi TimePrediction-DSDV mampu memitigasi sebagian besar dampaknya.



Gambar 17. Grafik *Packet Delivery Ratio* 100 node

Berdasarkan Gambar 11. Grafik di atas menunjukkan bahwa PDR menurun paling signifikan pada kepadatan 100 node, terutama pada kecepatan tinggi. TimePrediction-DSDV tetap menunjukkan peningkatan PDR sebesar 13.00%, 10.50%, dan 8.50% secara berturut-turut pada kecepatan 30 m/s, 60 m/s, dan 90 m/s. Pada kepadatan tinggi, peluang interferensi dan perubahan rute meningkat, tetapi TimePrediction-

DSDV mampu menjaga performa jaringan dengan rute yang lebih andal.

V. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan, algoritma TimePrediction-DSDV secara konsisten menunjukkan peningkatan kinerja dibandingkan DSDV standar pada berbagai skenario simulasi. Peningkatan *throughput* mencapai 15% pada 25 node, 14.5% pada 50 node, dan 13% pada 100 node, sementara penurunan *end-to-end delay* mencapai 22% pada 25 node, 20.8% pada 50 node, dan 19.5% pada 100 node. Untuk *Packet Delivery Ratio* (PDR), peningkatan masing-masing sebesar 12.5%, 11.6%, dan 10.67% pada jumlah node yang sama. Peningkatan kinerja ini lebih signifikan pada kecepatan node yang lebih rendah (30 m/s) dan kepadatan node yang lebih sedikit (25 node). Meskipun kinerja jaringan menurun seiring meningkatnya kecepatan dan kepadatan node, algoritma TimePrediction-DSDV tetap unggul dibandingkan DSDV standar dalam menjaga efisiensi dan stabilitas jaringan.

DAFTAR PUSTAKA

- [1] K. Hamzah, H. Al, P. H. Trisnawan, and R. A. Siregar, "Perbandingan Kinerja Protokol Routing HSR dan DSDV Pada Mobile Ad-Hoc Network (MANET)," vol. 4, no. 11, pp. 3880–3886, 2020.
- [2] C. S. Kumar, N. Sambasiva Rao, and G. Venkata Rami Reddy, "A study on routing metrics to improve the quality of service in MANET environment," *MATEC Web Conf.*, vol. 392, p. 01162, 2024.
- [3] S. Salah, R. Zaghal, and M. Abdeljawad, "A Mathematical-Based Model for Estimating the Path Duration of the DSDV Routing Protocol in MANETs," *J. Sens. Actuator Networks*, vol. 11, no. 2, 2022.
- [4] A. M. Eltahlawy, H. K. Aslan, E. G. Abdallah, M. S. Elsayed, A. D. Jurcut, and M. A. Azer, "A Survey on Parameters Affecting MANET Performance," *Electron.*, vol. 12, no. 9, 2023.
- [5] K. Nisar *et al.*, "QoS Analysis of the MANET routing protocols with Respect to *Delay, Throughput, & Network load*: Challenges and Open Issues," *14th IEEE Int. Conf. Appl. Inf. Commun. Technol. AICT 2020 - Proc.*, 2020.
- [6] R. Singh and N. Singh, "Performance Assessment of DSDV and AODV Routing Protocols in Mobile Adhoc Networks with Focus on Node Density and Routing Overhead," *2020 Int. Conf. Emerg. Smart Comput. Informatics, ESCI 2020*, pp. 298–303, 2020.
- [7] Q. Razouqi, A. Boushehri, M. Gaballa, L. Alsaleh, and M. Abbod, "Extended Comparison and Performance Analysis for Mobile Ad-Hoc Networks Routing Protocols Based on Different Traffic Load Patterns and Performance Metrics," *Electron.*, vol. 13, no. 14, 2024.
- [8] Z. L. G. Lala and I. R. Widiyari, "Analisis Performa Protokol Routing Proaktif dan Reaktif pada MANET dengan Menggunakan OMNeT++," *J. JTJK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 8, no. 1, pp. 22–29, 2024.
- [9] N. Amelia, Q. Aina, M. Husni, and T. Ciptaningtyas, "Simulasi dan Analisis Kinerja MANET pada Daerah Pasca Bencana dengan Protokol Routing Proaktif (Studi Kasus : Gempa Palu 2018)," vol. 13, no. 3, pp. 161–167, 2024.
- [10] H. Berradi, A. Habbani, N. Mouchfiq, and M. Souidi, "Improvement of olsr protocol using the hello message scheme based on neighbors mobility," *J. Commun.*, vol. 15, no. 7, pp. 551–557, 2020.
- [11] M. U. Wicaksono, "Analisis Pengaruh Penambahan Node Dalam Kinerja Routing Protocol Destination Sequence Distance Vector (DSDV) Pada Jaringan Vanet," 2020.
- [12] A. Agussalim, "Analisis Pengaruh Mobilitas Node Pada Kinerja Aodv Protokol Menggunakan Emulator Mininet-Wifi," *SCAN - J. Teknol. Inf. dan Komun.*, vol. 16, no. 3, pp. 26–30, 2021.
- [13] H. R. Sitorus, P. H. Trisnawan, and A. Basuki, "Pengaruh Model Mobilitas Terhadap Konsumsi Energi Protokol Routing Optimized Link State Routing (OLSR) Pada Mobile Ad Hoc Network (MANET)," *... Inf. dan Ilmu Komput. e-ISSN*, vol. 5, no. 1, pp. 183–190, 2021.