

**ANALISIS PENGARUH VARIAN MODEL PROPAGASI
TERHADAP KOMUNIKASI DATA PADA PROTOKOL ROUTING
TORA, MDART DAN ZRP DI JARINGAN VANET (STUDI KASUS
PETA JALAN KOTA PRAYA DAN BYPASS LOMBOK)**

Tugas Akhir
Untuk memenuhi sebagian persyaratan
Mencapai derajat Sarjana S-1 Program Studi Teknik Informatika



Oleh:

**PRIYA TAMA SUGARA
F1D 013 086**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MATARAM
2020**

TUGAS AKHIR

**ANALISIS PENGARUH VARIAN MODEL PROPAGASI TERHADAP
KOMUNIKASI DATA PADA PROTOKOL ROUTING TORA, MDART DAN ZRP
DI JARINGAN VANET (STUDI KASUS PETA JALAN KOTA PRAYA DAN
BYPASS LOMBOK)**

Oleh :

**PRIYA TAMA SUGARA
FID 013 086**

Telah diperiksa dan disetujui oleh Tim Pembimbing:

1. Pembimbing Utama



Andy Hidayat Jatmika, S.T., M.Kom.
NIP: 19831209 201212 1 001

Tanggal: 5 - 3 - 2020

2. Pembimbing Pendamping



Arivan Zubaldi, S.Kom., M.T.
NIP: 19860913 2015041 001

Tanggal: 4 - 3 - 2020

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Teknik
Universitas Mataram




Prof. Dr. Eng. I Gede Pasek Suta Wijaya, S.T., M.T.
NIP: 19731130 200003 1 001

TUGAS AKHIR

ANALISIS PENGARUH VARIAN MODEL PROPAGASI TERHADAP KOMUNIKASI DATA PADA PROTOKOL ROUTING TORA, MDART DAN ZRP DI JARINGAN VANET (STUDI KASUS PETA JALAN KOTA PRAYA DAN BYPASS LOMBOK)

Oleh:

PRIYA TAMA SUGARA
FID 013 086

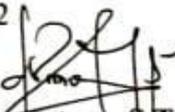
Telah dipertahankan di depan Dewan Penguji
Pada Tanggal 28 Februari 2020
Dan dinyatakan telah memenuhi syarat mencapai derajat sarjana S-1
Program Studi Teknik Informatika

Susunan Tim Penguji

1. Penguji 1

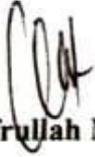

Dr. Eng. I Gde Putu Wirarama Wedashwara Wirawan, S.T., MT. Tanggal : 04 - 03 - 2020
NIP: 19840919 201803 1 001

2. Penguji 2


Fitri Bimantoro, S.T., M.Kom.
NIP: 19860622 201504 1 002

Tanggal : 04 - 03 - 2020

3. Penguji 3


Ahmad Zafrullah Mardiansyah, S.T., M.Eng.

Tanggal : 03 - 03 - 2020



Mataram, 06 Maret 2020
Dekan Fakultas Teknik
Universitas Mataram
Almarafudin, S.T., M.Sc., (Eng.), Ph.D.
NIP: 19681231 199412 1 001

PERNYATAAN KEASLIAN TUGAS AKHIR

Saya yang bertanda tangan di bawah ini bahwa dalam Skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Demikian surat pernyataan ini dibuat tanpa tekanan dari pihak manapun dan dengan kesadaran penuh terhadap tanggung jawab dan konsekuensi serta menyatakan bersedia menerima sanksi terhadap pelanggaran dan pernyataan tersebut.

Mataram, 05 Maret 2020

Priya Tama Sugara

PRAKATA

Puji syukur penulis panjatkan ke kehadiran Tuhan Yang Maha Pengasih dan Penyayang atas segala berkat, bimbingan, dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan Tugas Akhir dengan judul “Analisis Pengaruh Varian Model Propagasi Terhadap Komunikasi Data pada Protokol Routing TORA, MDART dan ZRP di Jaringan VANET (Studi Kasus Peta Jalan Kota Praya Dan Bypass Lombok)”

Tujuan dari Tugas Akhir ini adalah untuk menganalisis Pengaruh Varian Model Propagasi Terhadap Komunikasi Data pada Protokol Routing TORA, MDART dan ZRP di Jaringan VANET menggunakan NS-2 dengan parameter *Throughput*, *end to end Delay* (E2D) dan *Packet Delivery Ratio*.

Dalam penulisan skripsi ini tentunya tidak lepas dari kekurangan, baik aspek kualitas maupun aspek kuantitas dari materi penelitian yang disajikan. Semua ini didasarkan dari keterbatasan yang dimiliki penulis. Penulis menyadari bahwa skripsi ini jauh dari sempurna sehingga penulis membutuhkan kritik dan saran yang bersifat membangun untuk kemajuan penerapan tata kelola teknologi informasi di masa yang akan datang. Akhir kata semoga tidaklah terlampaui berlebihan, bila penulis berharap agar karya ini dapat bermanfaat bagi pembaca.

Mataram, 05 Maret 2020

Priya Tama Sugara

UCAPAN TERIMA KASIH

Tugas akhir ini dapat diselesaikan berkat bimbingan dan dukungan ilmiah maupun materil dari berbagai pihak, oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih yang setulus-tulusnya kepada :

1. Bapak Mazuni, Ibu Supartini, Saudari Debi Dwi Alsafiroh, Febi Tri Anggraini, Molly dan Saudara Prastiono.,ST serta keluarga besar selaku pemberi dukungan utama yang selalu memberikan do'a dan memberikan bimbingan dan dukungan dalam penyelesaian Tugas Akhir ini dengan baik.
2. Bapak Andy Hidayat Jatmika, ST.,M.Kom. selaku dosen pembimbing utama yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir ini, sehingga dapat terselesaikan dengan lancar.
3. Bapak Ariyan Zubaidi,S.Kom.,MT, selaku dosen pembimbing pendamping yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir ini, sehingga dapat terselesaikan dengan maksimal.
4. Bapak Dr.Eng. I Gde Putu Wirarama Wedashwara Wirawan, ST.,MT, Bapak Fitri Bimantoro.,ST.,M.Kom dan Bapak Ahmad Zafrullah Mardiansyah, ST.,M.Eng, selaku dosen penguji yang telah memberikan kritik dan saran yang bersifat membangun yang sangat diperlukan dalam penyelesaian Tugas Akhir ini.
5. Mahasiswa Teknik Informatika yang telah mendukung dengan memberikan semangat dan telah membantu penulis dalam proses penyelesaian Tugas Akhir ini.
6. Semua pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan bimbingan kepada penulis dalam menyelesaikan Tugas Akhir ini. Semoga Tuhan Yang Maha Esa memberikan imbalan yang setimpal atas bantuan yang diberikan kepada penulis.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
PERNYATAAN KEASLIAN TUGAS AKHIR	iv
PRAKATA	v
UCAPAN TERIMA KASIH	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
INTISARI	xii
ABSTRACT	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	5
2.1 Tinjauan Pustaka	5
2.2 Dasar Teori	7
2.2.1 Jaringan VANET (Vehicular Adhoc Network).....	7
2.2.1.1 Karakteristik VANET.....	8
2.2.2 Protokol Routing Pada VANET	9
2.2.3 Temporally Ordered Routing Algorithm (TORA)	9
2.2.4 Protokol Routing MDART	9
2.2.5 Zone Routing Protokol (ZRP)	10
2.2.6 Model Propagasi.....	10

2.2.6.1 Propagasi Free Space	10
2.2.6.2 Propagasi Two Ray Ground	11
2.2.6.3 Model Propagasi Shadowing.....	12
2.2.6.4 Model Propagasi Nakagami	13
2.2.7 Simulation of Urban Mobility (SUMO)	14
2.2.8 Network Simulator 2 (NS-2)	15
2.2.9 OpenStreetMap (OSM)	15
2.2.10 Java Open Street Map editor (JOSM).....	16
2.2.11 Parameter Pengukuran Kinerja Jaringan	16
BAB III METODE PENELITIAN.....	18
3.1 Identifikasi Kebutuhan Sistem	18
3.1.1 Kebutuhan Perangkat Keras	18
3.1.2 Kebutuhan Perangkat Lunak	18
3.2 Diagram Alir Penelitian.....	19
3.3 Perancangan Skenario Simulasi.....	20
3.4 Parameter Simulasi	20
3.5 Perancangan Peta Simulasi	21
3.6 Perancangan Simulasi Jaringan VANET	22
3.7 Parameter Uji atau Quality Of Servis (QOS).....	23
3.8 Jadwal Pelaksanaan Kegiatan	24
BAB IV METODE PENELITIAN	25
4.1 Menentukan Skenario.....	25
4.1.1 Mengatur Parameter Trafik	25
4.1.2 Mengunduh Peta.....	25
4.1.3 Mengedit Peta.....	26
4.1.4 Konversi File .OSM	27
4.1.5 Membuat File .SUMO Cfg.....	28
4.1.6 Membuat dan Menjalankan File TCL	29
4.1.7 Filtering Trace File Menggunakan Script AWK	32
4.2 Hasil Uji Coba	34
4.2.1 Analisis Nilai Rata-rata Hasil Simulasi.....	35

4.2.2 Hasil Simulasi di Jalan Kota Praya.....	37
4.2.2.1 Analisis Throughput	37
4.2.2.2 Analisis Packet Delivery Ratio (PDR)	39
4.2.2.3 Analisis Average End-To-End Delay	42
4.2.3 Hasil Simulasi di Baypass Lombok	44
4.2.3.1 Analisis Throughput	44
4.2.3.2 Analisis Packet Delivery Ratio (PDR)	46
4.2.3.3 Analisis Average End-To-End Delay	48
BAB V KESIMPULAN	54
5.1 Kesimpulan	54
5.2 Saran	55
DAFTAR PUSTAKA	

DAFTARGAMBAR

Gambar 2.1 Jaringan VANET	8
Gambar 3.1 Diagram Alir Penelitian	19
Gambar 3.2 Diagram Blok Pembuatan Peta dengan SUMO	22
Gambar3.3 Diagram Blok Simulasi.....	23
Gambar 4.1 Tampilan Saat Mengunduh Peta Jalan Kota Praya	26
Gambar 4.2 Tampilan Saat Mengunduh Peta Jalan Baypass Lombok	26
Gambar 4.3 Peta Kota Praya Yang Sudah di Rapikan.....	27
Gambar4.4 Peta Baypass Lombok Yang Sudah di Rapikan.....	27
Gambar 4.5 Throughput Protokol TORA.....	37
Gambar 4.6 Throughput Protokol MDART	38
Gambar 4.7 Throughput Protokol ZRP.....	39
Gambar4.8 Packet Delivery Ratio Protokol TORA.....	39
Gambar 4.9 Packet Delivery Ratio Protokol MDART	40
Gambar 4.10 Packet Delivery Ratio Protokol ZRP	41
Gambar 4.11 Delay Protokol TORA	42
Gambar4.12 Delay Protokol MDART.....	43
Gambar 4.13 Delay Protokol ZRP	43
Gambar 4.14 Throughput Protokol TORA	44
Gambar 4.15Throughput Protokol MDART	45
Gambar4.16Throughput Protokol TORA	45
Gambar 4.17 Packet Delivery Ratio Protokol TORA.....	46
Gambar4.18 Packet Delivery Protokol MDART	47
Gambar 4.19 Packet Delivery Protokol ZRP.....	47
Gambar 4.20 Delay Protokol TORA	48
Gambar 4.21 Delay Protokol MDART.....	49
Gambar4.22 Delay Protokol ZRP	49

DAFTAR TABEL

Tabel 2.1 Nilai Tipikal dari Eksponen Path Loss	13
Tabel 2.2 Nilai Khas dari Shadowing Deviation dengan dB	13
Tabel3.1 Spesifikasi Perangkat Keras yang digunakan	18
Tabel3.2 Skenario Perubahan Jumlah Node	20
Tabel3.3 Parameter Simulasi	21
Tabel3.4 Parameter Pengujian Jaringan.....	23
Tabel3.5 Rencana Pelaksanaan Penelitian.....	24
Tabel 4.1 Perintah Pengaturan Parameter Trafik.....	25
Tabel 4.2 Perintah Konversi Dengan Netconvert	28
Tabel4.3 Perintah Konversi Denga Randomtrips.py	28
Tabel4.4 Perintah Konversi Dengan Duarouter.....	28
Tabel4.5 Isi File .SUMO Cfg.....	28
Tabel4.6 Perintah Konversi File Dengan SUMO	29
Tabel4.7 Perintah Konversi Denga Traceexporter.py	29
Tabel 4.8 Isi Dari File TCL	30
Tabel 4.9 Perintah Menjalankan File TCL	32
Tabel4.10 Isi File AWK.....	32
Tabel4.11 Perintah Filtering Dengan File AWK.....	34
Tabel4.12 Hasil Simulasi di Jalan Kota Praya.....	35
Tabel4.13 Hasil Simulasi di Baypass Lombok	36

INTISARI

Vehicular Ad-hoc Network adalah bagian dari *Mobile Ad-hoc Network* (MANET), tetapi *node* pada VANET memiliki tingkat mobilitas yang lebih tinggi. Setiap *node* yang bergerak akan menyebabkan aktivitas jaringan (perubahan topologi) seperti pencarian rute dan pengiriman pesan ditangani oleh setiap *node*. Sehingga setiap *node* harus memiliki protokol *routing* untuk memudahkan komunikasi dalam jaringan. Pemilihan protokol *routing* yang sesuai diperlukan dalam skenario *urban* dan *rural*. Penelitian ini bertujuan untuk mengetahui pengaruh varian model propagasi pada komunikasi data pada protokol *routing* TORA, MDART dan ZRP pada jaringan VANET di peta jalan Kota Praya dan jalan *Bypass* Lombok dengan luas 1000x1000 m². Untuk variasi protokol *routing* menggunakan empat variasi model propagasi, yaitu model propagasi *freespace*, *tworayground*, *shadowing*, dan *nakagami*. Jumlah *node* yang digunakan dalam peta jalan Kota Praya adalah 35, 55, 70 *node* dan di jalan *Bypass* Lombok 25, 40, 60 *node*. Parameter yang digunakan adalah *throughput*, *packet delivery ratio* dan *end to end delay*. Dalam skenario *urban* dengan luas 1000x1000 m², hasilnya pada skenario urban nilai rata-rata yang terbaik pada protokol TORA parameter *throughput* sebesar 35,50 Kbps pada propagasi *nakagami*, dari skenario *rural* pada protokol MDART parameter PDR nilai rata-rata sebesar 99,54% pada propagasi *shadowing* dan untuk parameter *delay* sebesar 7,10 *second* pada propagasi *freespace*.

Kata kunci : VANET, TORA, MDART, ZRP, Propagasi, NS2

ABSTRACT

Vehicular Ad-hoc Network is part of the Mobile Ad-hoc Network (MANET), but the nodes on VANET have a higher level of mobility. Each node that moves will cause network activity (topology changes) such as route search and message sending handled by each node. So that each node must have a routing protocol to facilitate communication in the network. The selection of the appropriate routing protocol is needed in urban and rural scenarios. This study aims to determine the effect of propagation model variants on data communication on the TORA, MDART and ZRP routing protocols on the VANET network on the Praya City and Bypass Lombok road maps with an area of 1000x1000 m². For the variation of the routing protocol using four variations of the propagation model, namely the freespace propagation model, tworayground, shadowing, and nakagami. The number of nodes used in the Praya City road map is 35, 55, 70 nodes and in Lombok Bypass 25, 40, 60 nodes. The test parameters used are throughput, packet delivery ratio and end to end delay. In an urban scenario with an area of 1000x1000 m², the results in urban scenarios the best average value in the TORA protocol throughput parameter is 35.50 Kbps in the nakagami propagation, from the rural scenario in the MDART protocol PDR parameters the average value is 99.54% in the shadowing propagation and for the delay parameter is 7,10 seconds on freespace propagation.

Keywords: VANET, TORA, MDART, ZRP, Propagation, NS2

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi Informasi dan Komunikasi atau biasa disebut *Information Communication Technology* (ICT) telah banyak membantu produktivitas manusia yang semakin meningkat. Seiring dengan kemajuan teknologi informasi yang terus berkembang, perangkat komunikasi mengalami pengembangan, dimana pengguna dapat dengan mudah menggunakan perangkat komunikasi kapanpun dan dimanapun ketika kita membutuhkannya. Penggunaan perangkat bergerak ini bertujuan untuk memberikan kemudahan bagi pengguna untuk saling berkomunikasi dan bertukar data dengan mudah. Salah satu model dari perkembangan jaringan nirkabel yang dibutuhkan untuk memenuhi kebutuhan masyarakat ini adalah jaringan *Vehicular Ad-hoc Network* (VANET).

Jaringan VANET adalah jaringan dengan *node* berupa kendaraan. Kendaraan tersebut dapat berkomunikasi dengan kendaraan lainnya, maupun dengan berbagai piranti komunikasi yang ada di sepanjang jalan raya. VANET merupakan bagian dari *Mobile Ad-hoc Network* (MANET), namun *node* pada VANET memiliki tingkat mobilitas yang lebih tinggi. Setiap *node-node* yang bergerak akan menyebabkan aktifitas jaringan (perubahan topologi) misalnya pencarian rute dan pengiriman pesan di tangani oleh setiap *node*. Sehingga setiap *node* harus mempunyai *routing protocol* untuk memudahkan komunikasi dalam jaringan. Pemilihan *routing protocol* yang tepat ini diperlukan dalam skenario jalan kota (*urban*) dan pinggiran kota (*rural*).

Untuk memudahkan komunikasi dalam jaringan, sebuah *routing* protokol digunakan untuk menemukan dan mengatur rute antar *node*. Pemilihan *routing* protokol yang tepat diperlukan dalam skenario jalan kota (*urban*) dan pinggiran kota (*rural*). Pada jalan kota seperti yang kita ketahui banyak gedung dan pepohonan yang dapat mengakibatkan gangguan terhadap komunikasi antar kendaraan.

Terdapat tiga klasifikasi protokol jaringan yang digunakan yaitu protokol reaktif, proaktif dan *hybrid*. Tujuan dari protokol *routing* adalah untuk memiliki kecepatan yang efisien pembentukan antara sepasang *node*, sehingga pesan bisa disampaikan tepat pada waktunya. Dalam penerapannya, kinerja VANET juga sangat dipengaruhi oleh model

propagasi [1]. Suatu bagian penting dari setiap simulasi jaringan nirkabel adalah pilihan tentang model propagasi yang akan digunakan untuk memodelkan kinerja saluran jaringan nirkabel atau set saluran. Untuk mengetahui kinerja dari varian model propagasi tersebut menggunakan model propagasi seperti *freespace*, *tworayground*, *shadowing* dan *nakagami*. Model ini diperlukan dalam simulator untuk menghitung kekuatan sinyal transmisi nirkabel di stasiun penerima.

Simulasi protokol *routing* dibuat menggunakan *tools Network Simulator 2 (NS2) versi 2.35*, *Simulation for Urban Mobility (SUMO)* dan rute yang akan digunakan diambil dari situs *web OpenStreetMap*. Adapun *routing* protokol yang digunakan seperti protokol reaktif yaitu *temporally ordered routing algorithm (TORA)*, protokol proaktif seperti *multi-path dynamic addressing routing (MDART)* dan protokol *hybrid zone routing protocol (ZRP)*.

TORA merupakan protokol *routing* reaktif dan hasil yang didapatkan adalah protokol TORA Untuk kondisi jaringan yang padat dan kecepatan tinggi, *delay* dari protokol TORA menjadi lebih besar dibandingkan DSR dan AODV [2]. MDART adalah protokol proaktif berdasarkan hasil disimpulkan bahwa MDART berkinerja lebih baik dalam hal PDR karena mendapat nilai maksimum PDR di antara protokol DSR dan AODV [3]. ZRP merupakan protokol *routing* jenis *hybrid* memiliki performa yang baik dibandingkan dengan protokol *routing* GPSR dari segi *Packet Delivery Rasio (PDR)* dan *packet loss* pada jaringan VANET [4].

Penelitian ini diterapkan pada peta jalan Kota Praya dan *Bypass* Lombok, alasan dipilihnya peta jalan Kota Praya karena keadaan lalu lintas di jalan Kota Praya yang padat dapat menyebabkan terjadinya kemacetan sehingga sangat cocok sebagai objek penelitian, begitu juga di *Bypass* Lombok sangat cocok menjadi objek penelitian dikarenakan keadaan lalu lintasnya di pinggiran kota juga sangat padat karena jalur yang digunakan menuju ke Bandara Internasional Lombok. Dengan skenario perubahan jumlah *node* dan beberapa parameter yang akan dianalisis pada saat proses pengujian, diantaranya *throughput* yaitu jumlah rata-rata paket data yang terkirim oleh sumber, *packet delivery ratio* yaitu hasil perbandingan jumlah paket yang diterima oleh *node* yang menerima paket dengan total paket yang dikirimkan, dan *end to end delay (E2D)* merupakan waktu pengiriman paket data dari sumber ke penerima.

1.2 Rumusan Masalah

Permasalahan yang akan menjadi objek pada penelitian ini adalah bagaimana pengaruh varian model propagasi terhadap komunikasi data pada protokol *routing* TORA, MDART dan ZRP di jaringan VANET menggunakan NS-2 khususnya pada peta jalan Kota Praya dan *Bypass* Lombok?

1.3 Batasan Masalah

Penelitian ini memiliki beberapa batasan masalah agar penelitian dapat lebih fokus, dalam mencapai tujuan dan sasaran berdasarkan pada rumusan masalah diatas. Berikut adalah batasan masalah dari penelitian ini antara lain:

1. Jaringan yang digunakan adalah jaringan *vehicular ad-hoc network* (VANET).
2. *Network Simulator* yang digunakan adalah *network simulator 2* (NS2) versi 2.35.
3. *Mobility generator* yang digunakan adalah SUMO.
4. Jenis propagasi yang digunakan adalah *free space*, *tworayground*, *shadowing* dan *nakagami*.
5. *Routing* protokol yang digunakan seperti protokol reaktif (TORA), protokol proaktif (MDART) dan protokol *hybrid* (ZRP).
6. Simulasi mobilitas dilakukan berdasarkan skenario di lingkungan *urban* yaitu sekitar Kota Praya dan lingkungan *rural* di *Bypass* Lombok.
7. Analisis kinerja jaringan berdasarkan *throughput*, *packet delivery ratio* (PDR), dan *end to end delay* (E2D).
8. Aspek keamanan jaringan tidak dibahas.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui pengaruh varian model propagasi terhadap komunikasi data pada protokol *routing* TORA, MDART dan ZRP pada jaringan *Vehicular Ad-hoc Network* atau VANET khususnya pada peta jalan Kota Praya dan *Bypass* Lombok.

1.5 Manfaat Penelitian

Adapun manfaat setelah dilakukannya proses penelitian ini diharapkan dapat memberikan gambaran bagaimana pengaruh varian model propagasi terhadap komunikasi data pada protokol *routing* TORA, MDART dan ZRP di jaringan VANET pada peta jalan Kota Praya dan *Bypass* Lombok.

1.6 Sistematika Penulisan Penelitian

Sistematika penulisan Tugas Akhir ini dibagi menjadi beberapa bagian, yaitu :

1. PENDAHULUAN

Bab ini berisi latar belakang penulisan tugas akhir, rumusan masalah, batasan masalah, metodologi penelitian dan sistematika penulisan.

2. TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bab ini menjelaskan mengenai penelitian yang dilakukan sebelumnya dan teori yang berkaitan dengan judul atau masalah tugas akhir.

3. METODE PENELITIAN

Bab ini berisi perencanaan simulasi jaringan dan membahas langkah-langkah penelitian.

4. HASIL DAN PEMBAHASAN

Bab ini berisi pelaksanaan simulasi dan hasil analisis data simulasi jaringan.

5. KESIMPULAN DAN SARAN

Bab ini berisi beberapa kesimpulan yang didapat dan saran-saran berdasarkan hasil analisis data simulasi jaringan.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Pada penelitian [2] bertujuan untuk menganalisa performa dari protokol DSR, AODV, dan TORA dengan perbandingan *packet delivery ratio* (PDR), *end to end delay*, *routing load*, *packet loss percentage* (PLP), dan debit antara ketiga protokol *routing* tersebut. Hasil yang didapatkan adalah protokol TORA pada jaringan dengan jumlah *node* sedikit dan kecepatan gerak yang rendah menghasilkan *delay* yang paling sedikit. Untuk kondisi jaringan yang padat dan kecepatan tinggi, *delay* dari protokol TORA menjadi lebih besar dibandingkan DSR dan AODV, dari sisi pesat transfer atau debit, protokol DSR menghasilkan kinerja yang terbaik pada jaringan dengan jumlah *nodes* sedikit kinerja tersebut menurun seiring dengan kepadatan jaringan yang bertambah.

Pada penelitian [3] dilakukan pada jalan raya dengan perbandingan terhadap tiga macam protokol *routing* yaitu DSR, AODV dan MDART, hasilnya adalah dalam bentuk PDR dan *throughput*, dapat disimpulkan bahwa MDART berkinerja lebih baik dalam hal PDR karena mendapat nilai maksimum di antara protokol *routing* DSR dan AODV. Dan dalam hal *throughput* AODV berkinerja lebih baik dibandingkan protokol lainnya.

Pada penelitian [4] dilakukan pada peta dengan kondisi lingkungan *urban* (perkotaan) dengan perbandingan terhadap dua macam protokol *routing* yaitu ZRP dan GPSR pada jaringan VANET dengan berbagai parameter simulasi adalah menggunakan variasi jumlah *node* 20 sampai 100 *node* dengan kecepatan laju *node* adalah antara 30-50 km/h (*low speed*) dan 50-80 km/h (*high speed*) dengan luas area simulasi adalah 800 x 800 m². Pada penelitian ini model propagasi yang digunakan adalah *tworayground* dan simulasi dilakukan selama 200 detik, sedangkan untuk mengukur kinerja masing-masing protokol *routing* parameter pengujian yang digunakan adalah *throughput*, *hdr*, *end to end delay*, dan *packet loss*. Hasil kesimpulan pada penelitian ini menyatakan bahwa protokol ZRP memiliki kinerja yang lebih baik untuk pengiriman paket data, meskipun dengan nilai *throughput* yang rendah. Sementara itu GPSR kurang dapat diandalkan dalam pengiriman paket dari sumber ke tujuan, akan tetapi GPSR memiliki keunggulan dengan nilai *delay* yang rendah dan *throughput* yang tinggi.

Pada penelitian [5] dilakukan perbandingan terhadap kinerja beberapa variasi model propagasi yaitu diantaranya model propagasi *free space*, *tworayground*, *shadowing*, dan *nakagami*. Adapun parameter simulasi yang ditentukan pada penelitian tersebut adalah menggunakan area simulasi 1000x1000 m², jumlah variasi *node* yang dipakai adalah 500, 55, 75, 100, 125, 150, 175, dan 200, pada peta dengan kondisi lingkungan *urban* (perkotaan), dan menggunakan protokol *routing* AODV, waktu simulasi yang digunakan 100 detik, dan kecepatan pergerakan *node* adalah antara 8.33-13.89 meter/detik. Parameter pengujian yang digunakan untuk menguji kemampuan masing-masing model propagasi adalah *throughput*, *packet loss*, *transmitted packet*, dan *received packet*. Hasil yang diperoleh dari simulasi penelitian ini menunjukkan bahwa model propagasi *nakagami* memiliki performa yang paling baik dibandingkan ketiga model propagasi lainnya, baik itu berdasarkan nilai *throughput*, *packet loss*, *transmitted packet*, maupun *received packet*.

Pada penelitian [6] dilakukan perbandingan beberapa variasi model propagasi yaitu *free space*, *tworayground*, dan *shadowing* pada protokol *routing* AODV, pada peta dengan kondisi lingkungan *urban* (perkotaan), dengan parameter simulasi yang digunakan adalah kecepatan pergerakan *node* berada di kisaran antara 1 m/s sampai 40 m/s, jumlah *node* berkisar antara 5 hingga 40 *node*, dengan parameter ukur yaitu *packet delivery ratio* (pdr), *end to end delay* (E2D), dan *throughput*. Hasil yang didapat pada penelitian ini bahwa pada parameter ukur PDR model propagasi *shadowing* memiliki hasil yang baik, namun seiring dengan bertambahnya jumlah *node* maka model propagasi *free space* memiliki performa yang lebih baik. Untuk *throughput* rata-rata *shadowing model* memiliki nilai yang terbaik dari kedua model propagasi lainnya. Sedangkan untuk parameter ukur E2D model propagasi *free space* memiliki nilai yang baik diikuti dengan model *shadowing*.

Pada penelitian [7] dilakukan analisis terhadap kinerja model propagasi *tworayground*, *nakagami*, dan *shadowing* dengan parameter simulasi yang dilakukan adalah jumlah *node* sebanyak 12 *node/km/lane*, kecepatan gerak *node* 70-120 km/h pada peta dengan kondisi lingkungan perkotaan (*urban*). Penelitian ini melakukan beberapa skenario simulasi yaitu, pada skenario pertama menggunakan sebuah *node* atau *vehicle* yang melakukan *broadcast* paket data ke *node* lainnya terlihat perbedaan hasil yang dihasilkan oleh beberapa model propagasi. Kemudian pada skenario percobaan kedua menggunakan parameter uji seperti skenario pertama akan tetapi *node-node*

mengirimkan status status mereka secara *continue*, sedangkan salah satu *node* akan bertugas sebagai *broadcaster* paket data kepada *node* yang lainnya, hasilnya adalah model propagasi *tworayground* mengalami waktu *delay* yang terbesar bila dibandingkan dengan model propagasi lainnya (*shadowing* dan *nakagami*).

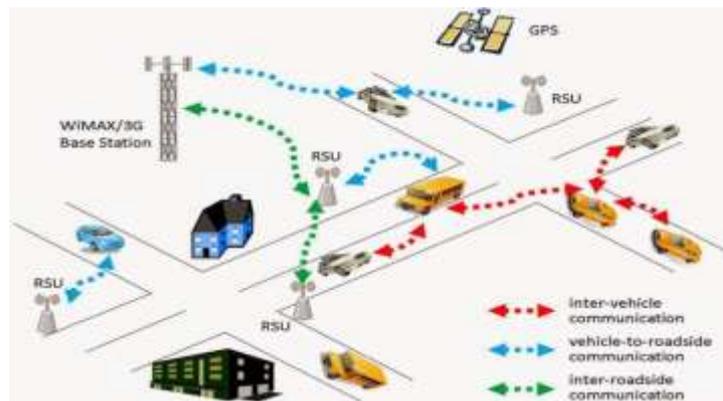
2.2 Dasar Teori

Pada sub bab ini akan dibahas tentang landasan teori yang berkaitan dengan penelitian yang akan dikerjakan dalam lingkup jaringan VANET.

2.2.1 Jaringan VANET (*Vehicular Ad Hoc Network*)

Sejumlah kendaraan yang terhubung satu sama lain melalui sebuah rangkaian *ad hoc* membentuk suatu jaringan nirkabel yang disebut *Vehicular Ad hoc Network*. *Vehicular Ad Hoc Network* (VANET) merupakan kategori khusus dari *mobile ad hoc networks* (MANET's), ditandai dengan mobilitas tinggi dan konektivitas yang rendah. VANET juga dapat dikatakan sebagai sebuah jaringan terorganisir yang dibentuk dengan menghubungkan antar kendaraan dan RSU (*Road Side Unit*). Lebih lanjut RSU terhubung ke jaringan *backbone* berkecepatan tinggi melalui koneksi jaringan. Kepentingan peningkatan baru-baru ini telah diajukan pada aplikasi melalui komunikasi V2V (*Vehicle to Vehicle*) dan V2I (*Vehicle to Infrastructur*), bertujuan untuk meningkatkan keselamatan mengemudi dan manajemen lalu lintas sementara bagi para pengemudi dan penumpang. Dalam VANET, RSU (*Road Side Unit*) dapat memberikan bantuan dalam menemukan fasilitas seperti restoran dan pom bensin serta mem-*broadcast* pesan yang terkait seperti kecepatan kendaraan kepada pengendara lain. Sebagai contoh sebuah kendaraan dapat terhubung dengan lampu lalu lintas melalui komunikasi V2I dan lampu lalu lintas dapat memberikan informasi ke kendaraan ketika dalam keadaan lampu ke kuning atau merah. Ini dapat berfungsi sebagai tanda pemberitahuan kepada pengemudi dan akan sangat membantu para pengendara ketika mereka sedang berkendara selama kondisi cuaca buruk atau di daerah asing. Hal ini dapat mengurangi terjadinya kecelakaan. Melalui komunikasi V2V, pengendara bisa mendapatkan informasi yang lebih cepat dan lebih baik serta mengambil tindakan awal untuk menghadapi situasi yang abnormal. Untuk mencapai hal ini, suatu OBU (*On-Board Unit*) secara teratur menyiarkan pesan yang terkait dengan informasi dari posisi

pengendara, waktu sekarang, arah pengemudian, kecepatan, status rem, lampu sen, percepatan atau perlambatan, kondisi jalan [8].



Gambar 2.1 Jaringan VANET [3]

2.2.1.1 Karakteristik VANET

VANET memiliki beberapa karakteristik yang lebih unggul dibanding dengan jaringan *Ad Hoc* lainnya, seperti :

1. Topologi Dinamis Tingkat Tinggi Perubahan topologi pada VANET disebabkan oleh pergerakan dari kendaraan dengan kecepatan tinggi.
2. Sering Terputusnya Jaringan Hasil dari topologi dinamis dapat diamati bahwa pemutusan sering terjadi antara dua kendaraan ketika sedang bertukar informasi. Pemutusan ini akan terjadi pada *sparse network*.
3. Pemodelan Mobilitas Pola mobilitas kendaraan tergantung pada lingkungan lalu lintas, jalan terstruktur, kecepatan kendaraan, perilaku mengemudi dan sebagainya.
4. Daya Baterai dan Kapasitas Penyimpanan Dalam kendaraan modern daya baterai dan penyimpanan tidak terbatas. Demikian yang terjadi pada MANET telah cukup daya untuk melakukan komputasi. Hal ini berguna untuk komunikasi yang efektif dan membuat rute keputusan.
5. Komunikasi Lingkungan antara kendaraan yang berbeda dalam *sparse network* dan *dense network*. Dalam membangun *dense network*, pohon dan benda lainnya berperan sebagai hambatan dan *sparse network* seperti jalan raya tidak hadir. Jadi *routing* yang digunakan untuk *sparse network* dan *dense network* akan berbeda.

6. Interaksi dengan Sensor Posisi saat ini dan pergerakan *node* dengan mudah dapat dirasan oleh sensor seperti perangkat GPS. Ini membantu untuk komunikasi yang efektif dan keputusan rute.

2.2.2 Protokol Routing pada VANET

VANET dengan jenis komunikasi *Vehicle to vehicle Communication* merupakan jaringan *wireless* yang berdiri tanpa infrastruktur, ini menjadikan pengiriman data pada VANET dilakukan dengan cara *estafet* dari *node* sumber menuju *node* tujuan. Mekanisme terkirimnya data pada jaringan disimpan pada protokol, begitu juga pada VANET, penggunaan protokol yang tepat pada VANET sangat berpengaruh pada kinerja jaringan yang ada.

Secara umum terdapat beberapa kategori protokol *routing*, yaitu protokol *proaktif*, *reaktif*, dan *hybrid*. Protokol *routing* proaktif memiliki ciri dimana tiap *node* yang ada di dalam jaringan selalu memperbarui tabel *routing* yang dimiliki dalam interval waktu tertentu, hingga setiap *node* yang terdapat dalam jaringan tersebut mengetahui semua rute menuju *node* yang lainnya. Pada protokol *reaktif* rute menuju *node* lain hanya akan dicari apabila terdapat permintaan dari *node* lainnya, dan isi dari tabel *routing*-nya adalah informasi rute menuju *node* tujuan saja. Sedangkan protokol *hybrid* adalah jenis protokol yang menggabungkan mekanisme kerja protokol *reaktif* dan *proaktif* [6].

2.2.3 Temporally Ordered Routing Algorithm (TORA)

Temporally Ordered Routing Algorithm (TORA) adalah *routing protocol* yang memiliki konsep *link reversal*. Protokol *routing* TORA dalam pengiriman paket dapat menyediakan banyak rute yang bisa digunakan oleh *node* pengirim untuk mengirimkan paketnya. *Node* pengirim menyediakan beberapa rute menuju *node* tujuan, sehingga jika satu rute gagal, maka dapat menggunakan rute lain. Dengan adanya banyak rute dari *node* pengirim, pengiriman paket data tidak akan terganggu saat pertama kali terjadinya perubahan jaringan.

2.2.4 Protokol Routing MDART

MDART adalah protokol *routing* reaktif yang memperbarui DART dalam menemukan rute yang *available* antara tujuan dengan titik asal. Inti dari protokol MDART ini terletak pada peningkatan informasi dalam *routing* disimpan dalam tiap

node sehingga ketika ada perubahan dalam suatu jalur, maka protokol ini akan mencari jalur lain dengan cepat.

2.2.5 Zone Routing Protocol (ZRP)

ZRP merupakan protokol *routing hybrid* dengan komponen *proactive* dan *reactive*. *Proactive routing* melakukan *routing* dengan *node* tujuan yang berada di dalam zona lokal. Sedangkan *reactive routing* melakukan *routing* menuju *node* tujuan yang berada di luar zona lokal.

ZRP mengelompokkan kumpulan *node* yang berada dalam suatu zona lokal dengan zona sentral menjadi pusat. Zona lokal tersebut dibatasi oleh suatu parameter yang disebut *hop*. Semua *node* yang berjarak sebanyak jumlah *hop* dari *node* sentral, berarti masuk ke dalam zona *routing* dari *node* sentral tersebut.

2.2.6 Model Propagasi

Salah satu bagian penting dalam jaringan VANET adalah pemilihan model propagasi yang akan digunakan untuk memprediksi besar kecilnya nilai daya sinyal yang diterima pada jarak tertentu dari pemancar. Lapisan fisik setiap *node* mempunyai nilai ambang batas daya minimal penerimaan sinyal. Jika paket diterima dengan kondisi dibawah nilai ambang batas, maka paket tersebut dianggap sebagai paket yang rusak. Berikut merupakan berbagai model propogasi.

2.2.6.1 Propagasi Free Space

Model Propagasi *Free Space* digunakan untuk memperkirakan kekuatan sinyal yang diterima ketika *transmitter* dan *receiver* tidak memiliki penghalang antara mereka (*clear, unobstructed line-of-sight*). Contoh komunikasi yang menggunakan propagasi ini adalah komunikasi satelit dan *microwave line-of-sight radio*. Dengan penggunaan propagasi ini pada skala yang besar, propagasi ini memprediksikan bahwa kekuatan *power* yang diterima menurun sejalan dengan kenaikan *power* pada jarak antara *transmitter* dan *receiver* (*T-R separation*). *Powerfree space* diterima oleh antena *receiver* yang dipisahkan dari antena *transmitter* disimbolkan dengan *d*, yang diberikan seperti pada persamaan 2.1.

$$Pr(d) = P_t G_t G_r \lambda^2 / ((4\pi)^2 d^2 L) \quad (2.1)$$

Dimana :

$Pr(d)$ = power yang diterima dimana merupakan fungsi dari *T-R separation*

Pt = power yang ditransmisikan

Gt = tegangan antena pada *transmitter*

Gr = tegangan antena pada *receiver*

d = jarak *T-R separation* dalam meter,

L = *loss factor* sistem yang tidak berhubungan dengan propagasi ($L \geq 1$)

$\lambda = c/f = 2\pi c/\omega c$

f = frekuensi

dimana, Pt adalah *power* yang ditransmisikan, $Pr(d)$ adalah *power* yang diterima dimana merupakan fungsi dari *T-R separation*, Gt adalah tegangan antena pada *transmitter*, d adalah jarak *T-R separation* dalam meter, L adalah *loss factor* sistem yang tidak berhubungan dengan propagasi ($L \geq 1$), dan X adalah panjang gelombang dalam meter. Dimana $\lambda = c/f = 2\pi c/\omega c$, f adalah frekuensi *carrier* dalam *Hertz*, ωc adalah frekuensi *carrier* dalam radian per detik, dan c adalah kecepatan cahaya dalam meter per detik. Nilai Pt dan Pr harus memiliki satuan yang sama sedangkan Gt dan Gr tidak memiliki satuan. Nilai *losses* L ($L \geq 1$) biasanya dikarenakan adanya atenuasi pada jalur transmisi, *filter losses* dan antena *losses* dalam sistem komunikasi. Nilai $L = 1$ mengindikasikan tidak adanya *loss* dalam sistem perangkat keras. *Model free space* biasanya merepresentasikan daerah komunikasi yang berada disekitar / lingkaran transmitter. Jika *receiver* berada di lingkaran tersebut, semua paket akan diterima. Selain itu, paket tersebut akan tidak mungkin diterima [8].

2.2.6.2 Propagasi Two Ray Ground

Dalam *mobile radio channel*, *single direct path* antara *base station* dan *mobile* terkadang hanya peralatan fisik biasa untuk propagasi dan rumus pada *free space* kurang akurat jika dalam penggunaannya berdiri sendiri. Model propagasi *Two Ray Ground* merupakan model yang berguna karena berdasar pada optik geometri dan dapat digunakan untuk *direct path* dan refleksi dari *ground* antara *transmitter* dan *receiver*. Model ini dirasa sangat akurat untuk memperkirakan kekuatan sinyal dalam skala luas dengan jarak beberapa kilometer untuk sistem *mobile radio* dengan menggunakan menara yang tinggi. *Power* yang diterima dengan jarak d diberikan pada persamaan 2.2.

$$Pr(d) = Pt Gt Gr h_t^2 h_r^2 / d^4 L \quad (2.2)$$

Dimana :

$P_r(d)$ = power yang diterima dimana merupakan fungsi dari *T-R separation*

P_t = power yang ditransmisikan

G_t = tegangan antena pada *transmitter*

G_r = tegangan antena pada *receiver*

h_t = tinggi dari antena *transmitter*

h_r = tinggi dari antena *receiver*

d = jarak *T-R separation* dalam meter

L = *loss factor* sistem yang tidak berhubungan dengan propagasi ($L \geq 1$)

dimana h_t dan h_r adalah tinggi dari antena *transmitter* dan *receiver*, nilai L diasumsikan sama dengan nilai L pada propagasi *free space*, $L=1$. Untuk parameter yang lain, masih sama dengan parameter pada propagasi *free space*. Berdasarkan rumus matematika diatas, *power loss* lebih cepat hilang dibandingkan dengan rumus matematika pada model propagasi *free space* ketika jaraknya bertambah. Namun, Model ini tidak memberikan hasil yang baik untuk jarak yang terlalu dekat dikarenakan osilasi yang disebabkan oleh konstruktif dan destruktif yang merupakan kombinasi dari model ini [8].

2.2.6.3 Model Propagasi *Shadowing*

Model propagasi *Shadowing* terdiri dari dua bagian, yang pertama dikenal sebagai model *pathloss*, yaitu memprediksi daya rata-rata yang diterima pada jarak d , didefinisikan sebagai $\overline{P_r(d)}$. Persamaan menggunakan jarak d_0 sebagai referensi. $\overline{P_r(d)}$ dihitung relatif terhadap $p_r(d_0)$ dinyatakan dengan persamaan 2.3.

$$\frac{P_r(d_0)}{\overline{P_r(d)}} = \left(\frac{d}{d_0}\right)^\beta \quad (2.3)$$

Dimana :

$P_r(d)$ = power yang diterima dimana merupakan fungsi dari *T-R separation*

d = jarak *T-R separation* dalam meter

β = sebagai eksponen *pathloss*

Yang dimana β disebut sebagai eksponen *pathloss*, dan biasanya nilai tersebut secara empiris ditentukan oleh pengukuran di lapangan (area simulasi). Dalam tabel 2.1 diberikan beberapa nilai khas dari parameter β berdasarkan tipikal lingkungan.

Tabel 2.1 Nilai tipikal dari eksponen *pathloss* β

Lingkungan		β
Outdoor	<i>Free Space</i>	2
	<i>Shadowing Urban Area</i>	2.7 sampai 5
Indoor	<i>Line-of-sight</i>	1.6 sampai 1.8
	<i>Obstructed</i>	1 sampai 6

Pathloss dalam dB dapat dihitung menggunakan persamaan 2.4.

$$\left[\frac{P_r(d)}{P_r(d_0)} \right]_{dB} = -10\beta \log \left(\frac{d}{d_0} \right) \quad (2.4)$$

Pada bagian kedua dari model propagasi *Shadowing* menunjukkan variasi daya yang diterima pada jarak yang pasti. Menggunakan variabel acak *log normal*, yaitu variabel distribusi Gaussian jika dikur dalam dB. Model propagasi *Shadowing* secara keseluruhan dapat didefinisikan dengan persamaan 2.5.

$$\left[\frac{P_r(d)}{P_r(d_0)} \right]_{dB} = -10\beta \log \left(\frac{d}{d_0} \right) + X_{dB} \quad (2.5)$$

Dimana X_{dB} adalah variable acak Gaussian dengan *mean* nol dan standar deviasi σ_{dB} . σ_{dB} dikenal sebagai *Shadowing deviation*. Pada tabel 2.2 menunjukkan beberapa nilai khas dari σ_{dB} .

Tabel 2.2 Nilai khas dari *Shadowing deviation* dengan dB

Lingkungan	σ_{dB} (dB)
<i>Outdoor</i>	4 sampai 12
<i>Office, HardPartition</i>	7
<i>Office, Soft Partition</i>	9.6
<i>Factory, Line-of-sight</i>	3 sampai 6
<i>Factory, Obstructed</i>	6.7

2.2.6.4 Model Propagasi Nakagami

Model propagasi *Nakagami* merupakan model yang bersifat matematis umum yang diterapkan pada berbagai kondisi *fading*. Dibandingkan dengan Model *TwoRayGround* dan *Shadowing*, model propagasi *Nakagami* memiliki parameter yang

dapat dikonfigurasi agar lebih cocok dan memungkinkan untuk direpresentasikan ke dalam *wireless communication channel*. Model propagasi *Nakagami* mampu memodelkan dari sebuah saluran yang bebas hambatan (*Free Space*), ke saluran yang lebih sedang di jalan raya, bahkan ke saluran yang lebih dramatis sekalipun seperti pada lingkungan perkotaan (*urban*). Model propagasi *Nakagami* untuk berbagai kondisi *fading* dapat ditentukan dengan parameter- m . Distribusi *Nakagami* memiliki *probability destiny function* (PDF) yang didefinisikan pada persamaan 2.6.

$$P_r(r) = \frac{2m^m x^{2m-1}}{\Gamma(m)\Omega^m} \exp\left[-\frac{mx^2}{\Omega}\right], x \geq 0, \Omega > 0, m \geq \frac{1}{2} \quad (2.6)$$

Dimana:

m = parameter *fading*, $m \geq 0.5$ sampai ~ (interger positif)

r = *amplitude fading*; Γ = fungsi gamma; $\Omega = E[R^2]$

2.2.7 Simulation of Urban Mobility (SUMO)

SUMO merupakan paket simulasi lalu lintas yang bersifat *open-source* yang memiliki kelengkapan fitur dan permodelannya, termasuk kemampuan jalannya jaringan untuk membaca format yang berbeda, permintaan dengan skala besar hingga pengaturan *routing* dan semua yang berhubungan dengan penelitian pergerakan dalam lalu lintas yang berfokus pada daerah penduduk padat (*urban*) [9]. SUMO terdiri dari beberapa macam *tools* yang digunakan dalam membangun simulasi lalu lintas pada tahap yang berbeda, diantaranya yaitu :

- *Netconvert*, berfungsi untuk melakukan konversi dari peta seperti *OpenStreetMap* menjadi format mentah SUMO.
- *Netgenerate*, berfungsi untuk membuat peta berbentuk seperti *grid*, *spider* dan bahkan *random network*. Sebelum melakukan proses *netgenerate*, pengguna dapat membuat *traffic light* pada peta dan menentukan kecepatan maksimum jalan. Hasil dari *netgenerate* ini berupa *file* dengan ekstensi (.net.xml).
- *randomTrips.py*, berfungsi membuat rute *random* yang akan dilalui oleh kendaraan dalam simulasi.
- *route2trips.py*, berfungsi membuat detail perjalanan setiap kendaraan berdasarkan *output* dari *randomTrips.py*.
- *duarouter*, *tool* yang berfungsi melakukan perhitungan rute berdasarkan definisi yang diberikan dan memperbaiki kerusakan rute.

- *sumo-gui*, GUI untuk melihat simulasi yang dilakukan oleh SUMO secara grafis.
- *traceExporter.py*. *tool* yang berfungsi untuk mengkonversi *output* dari sumo menjadi format yang dapat digunakan pada *simulator* lain.

2.2.8 *Network Simulator 2 (NS-2)*

Network Simulator 2 (NS-2) adalah suatu interpreter yang *object-oriented*, dan *discrete event-driven* yang dikembangkan oleh University of California Berkeley dan USC ISI sebagai bagian dari proyek *Virtual Internet Testbed (VINT)*. *Network Simulator* ini menjadi salah satu alat simulasi yang sangat berguna untuk menunjukkan simulasi jaringan melibatkan *Local Area Network (LAN)*, *Wide Area Network (WAN)*, dan dengan berkembangnya zaman berkembang pula fungsi dari alat simulasi ini, yaitu selama beberapa tahun belakangan ini alat simulasi ini melibatkan pada jaringan nirkabel (*wireless*) serta pada jaringan *ad hoc*. Beberapa keuntungan menggunakan *network simulator* sebagai perangkat lunak simulasi yaitu *network simulator* dilengkapi dengan *tools* validasi, pembuatan simulasi dengan menggunakan *network simulator* jauh lebih mudah daripada menggunakan *software developer* seperti Delphi atau C++, *network simulator* bersifat *open source* di bawah GPL (*Gnu Public License*) dan dapat digunakan pada sistem operasi windows dan sistem operasi Linux [1]. Versi terbaru dari NS-2 adalah ns-2.35 yang dirilis pada tahun 2011.

2.2.9 *OpenStreetMap (OSM)*

OpenStreetMap (OSM) merupakan sebuah proyek gabungan untuk membuat contoh peta dunia yang dapat dengan bebas diubah oleh siapapun. Dua buah faktor pendukung dalam pembuatan dan perkembangan OSM adalah kurangnya ketersediaan dari informasi peta mengenai sebagian besar daerah di dunia dan munculnya alat navigasi portabel yang terjangkau. Contoh utama dalam informasi geografis yang diberikan secara bebas. Hal yang paling penting adalah peta OSM dapat disimpan di dalam internet, dan siapapun dapat mengakses karena setiap menit terdapat pembaharuan pada peta OSM. Hal ini terjadi karena lebih dari dua juta yang terdaftar *editor OSM* yang terus-menerus membuat perubahan peta di seluruh dunia [8].

2.2.10 Java OpenStreetMap Editor (JOSM)

Java OpenStreetMap Editor (JOSM) merupakan sebuah aplikasi *desktop* untuk menyunting data pada *OpenStreetMap*. JOSM tidak membutuhkan koneksi Internet untuk menyunting. Aplikasi JOSM dapat diunduh pada alamat web berikut (<https://josm.openstreetmap.de/>). Penulis menggunakan aplikasi ini untuk menyunting dan merapikan potongan peta yang diunduh dari *OpenStreetMap*. Ada 3 mode operasi utama yang digunakan ketika pengeditan di JOSM, diantaranya :

- a. *Select* : untuk memilih elemen atau objek, edit atau melihat *tag* atau atribut.
- b. *Add* : untuk menambahkan titik baru yang bertujuan membuat jalan/fasilitas umum baru, dan memperpanjang jalan yang ada.
- c. *Delete* : untuk menghapus elemen atau objek.

2.2.11 Parameter Pengukuran Kinerja Jaringan

Parameter *QoS* merupakan metode pengukuran untuk mengetahui tingkat kinerja atau kualitas dari suatu jaringan dilihat dari hasil kolektif dan beragam performansi yang digunakan sebagai patokan. Beberapa parameter yang akan digunakan untuk menguji kinerja pada penelitian ini yaitu sebagai berikut:

- a. *Average Throughput*

Throughput merupakan suatu istilah yang mendefinisikan banyaknya bit yang diterima dalam selang waktu tertentu dengan satuan bit per *second* yang merupakan kondisi data *rate* sebenarnya dalam suatu jaringan. Secara umum *throughput* dinyatakan dalam persamaan 2.7.

$$T = \frac{Pr}{t} \times \text{ukuran paket} \quad (2.7)$$

Dimana :

T = *throughput* (bps)

Pr = jumlah paket yang diterima

t = total waktu pengamatan (s)

- b. *Packet delivery ratio (PDR)*

Packet delivery ratio merupakan perbandingan antara paket data yang berhasil diterima oleh *node* tujuan dengan total paket data yang dikirimkan oleh *node* sumber. *Packet Delivery Ratio* akan menurun pada saat kecepatan gerak *node* bertambah besar. Hal ini disebabkan bertambahnya kecepatan gerak *node* maka akan sering terjadi

perubahan posisi *node* sehingga rute rusak, menyebabkan hilangnya paket data yang lewat (*drop*). Secara umum *packet delivery ratio* dapat dilihat pada persamaan 2.8.

$$PDR (\%) = \frac{\Sigma \text{Paket diterima}}{\Sigma \text{Paket dikirim}} \times 100\% \quad (2.8)$$

c. *Average End-to-end delay*

Delay merupakan waktu pengiriman paket data dari sumber ke penerima, *delay* akan meningkat ketika kecepatan gerak *node* bertambah besar. Hal ini disebabkan pada saat kecepatan tinggi, maka kondisi jaringan akan semakin tidak stabil. Hal ini akan mengakibatkan semakin banyak paket RREQ yang dikirimkan, sehingga akan mengakibatkan peluang tabrakan antar paket semakin besar. Pada saat *node* bergerak dengan kecepatan tinggi, *route* semakin tidak tersedia sehingga paket data akan tertahan di *buffer* sementara sampai *route* baru di temukan. Hal ini akan mengakibatkan rata-rata *delay* pengiriman paket data akan semakin besar. Secara umum *average end-to-end delay* dapat dilihat pada persamaan 2.9.

$$\text{averageDelay}(m/s) = \frac{\text{total delay}}{\text{paketyangditerima}} \quad (2.9)$$

BAB III

METODE PENELITIAN

Perancangan merupakan bagian penting dari pembuatan suatu simulasi yang isinya berupa alur-alur dari proses pembuatan suatu simulasi. Sehingga bab ini secara khusus akan menjelaskan perancangan simulasi yang akan di buat.

3.1 Identifikasi Kebutuhan Sistem

Penelitian ini memiliki beberapa kebutuhan, yaitu perangkat keras yang digunakan pada penelitian ini adalah sebuah laptop dengan spesifikasi yang akan dirincikan pada sub bab 3.1.1 Spesifikasi perangkat keras, sedangkan untuk kebutuhan perangkat lunak merupakan sistem operasi dan aplikasi-aplikasi pendukung dalam penelitian yang akan dirincikan pada sub bab 3.1.2 Spesifikasi perangkat lunak.

3.1.1 Kebutuhan Perangkat Keras

Uji coba dilakukan dengan menggunakan Laptop ASUS X441S dengan spesifikasi perangkat yang dapat dilihat pada Tabel 3.1 dan parameter lingkungan uji coba penelitian yang digunakan pada NS-2 dapat dilihat pada Tabel 3.1.

Tabel 3.1 Spesifikasi perangkat keras yang digunakan

Komponen	Spesifikasi
CPU	Intel® Core N3060 CPU @ 2.48 GHz
Sistem Operasi	Linux Ubuntu 16.04 LTS 32-bit
Memori	2,00 GB
<i>Harddisk</i>	500 GB

3.1.2 Kebutuhan Perangkat Lunak

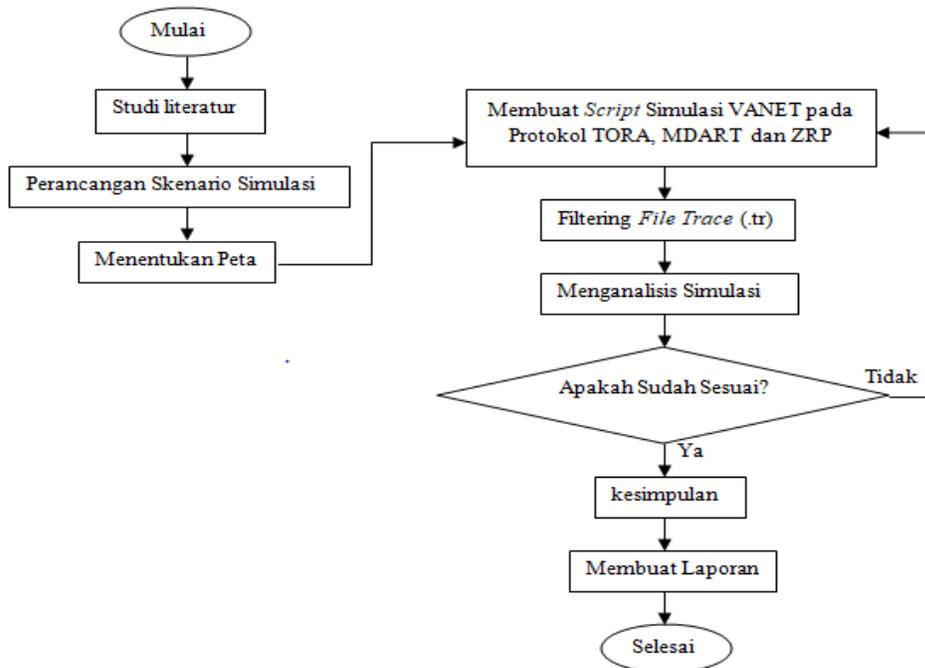
Adapun perangkat lunak yang digunakan pada penelitian ini adalah sebagai berikut :

- Sistem operasi Linux Ubuntu 16.04 LTS 32 bit
- *Network Simulator 2* (NS2) versi 2.35 untuk melakukan simulasi VANET.

- *Simulation of Urban Mobility* (SUMO) untuk membuat skenario mobilitas VANET.
- *Java OpenStreetMap* (JSOM) untuk menyunting peta yang didapat dari *OpenStreetMap*
- *Microsoft Excel* untuk membuat grafik hasil uji coba

3.2 Diagram Alir Penelitian

Pada penelitian ini, simulasi dilakukan di jalan Kota Praya dan *Bypass* Lombok. Dan pada tahap ini dilakukan pengkajian dari beberapa penelitian yang telah dilakukan sebelumnya. Pengkajian dilakukan terhadap berbagai jurnal dan *paper* mengenai performansi dari *routing* protokol TORA, MDART dan ZRP pada VANET dengan menggunakan *Network Simulator 2*, SUMO, dan aplikasi pendukung jaringan lainnya. Menghasilkan *output* berupa *trace file* (*.tr), selanjutnya *trace file* akan diuji pada tahap pengujian menggunakan parameter uji *throughput*, *packet delivery ratio* (PDR), dan *end to end delay* (E2D). Setelah pengujian akan dilakukan analisis dan ditarik suatu kesimpulan serta pembuatan laporan. Gambar diagram Alir simulasi pada penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian

3.3 Perancangan Skenario Simulasi

Pada penelitian ini proses perancangan skenario simulasi menggunakan VANET dengan unsur komunikasi antar kendaraan atau *inter vehicle communication*, yang membandingkan kinerja dari protokol TORA, MDART dan ZRP dengan melakukan 3 tahapan, yaitu tahap menentukan parameter simulasi, perancangan peta simulasi, dan perancangan alir simulasi menggunakan NS2.

3.4 Parameter Simulasi

Untuk mengetahui pengaruh dari perubahan jumlah *node*, dilakukan skenario perubahan jumlah *node* dalam jaringan, ini dilakukan agar dapat melihat performa dari kinerja protokol *routing* TORA, MDART dan ZRP dalam menangani perubahan topologi jaringan akibat perubahan jumlah *node*. Jumlah kepadatan kendaraan atau *node* ini terbagi menjadi tiga tingkatan kepadatan yakni tinggi, sedang, dan rendah. Kepadatan tinggi diambil dari hasil observasi dengan jumlah kendaraan yang paling banyak, kepadatan sedang diambil dari hasil observasi dengan jumlah kendaraan dengan nilai kepadatan berada pada posisi pertengahan kepadatan, dan nilai kepadatan rendah diambil dari hasil observasi kendaraan dengan jumlah kendaraan paling sedikit. Pengumpulan data dilakukan kurang lebih selama 1 minggu dengan mendata jumlah kendaraan roda 4 yang melintas pada Jalan Kota Praya dan *Bypass* Lombok selama 300 detik. Skenario perubahan jumlah *node* dapat dilihat pada Tabel 3.2.

Tabel 3.2 skenario perubahan jumlah *node*

Skenario	Perubahan jumlah <i>node</i> (tingkat kepadatan)			Kecepatan <i>node</i>
	Tinggi	Sedang	Rendah	
Jalan Kota Praya	70	55	35	50 km/jam 60 km/jam 80 km/jam
<i>Bypass</i> Lombok	60	40	25	60 km/jam 80 km/jam 100 km/jam

Adapun parameter simulasi selengkapnya seperti pada Tabel 3.3

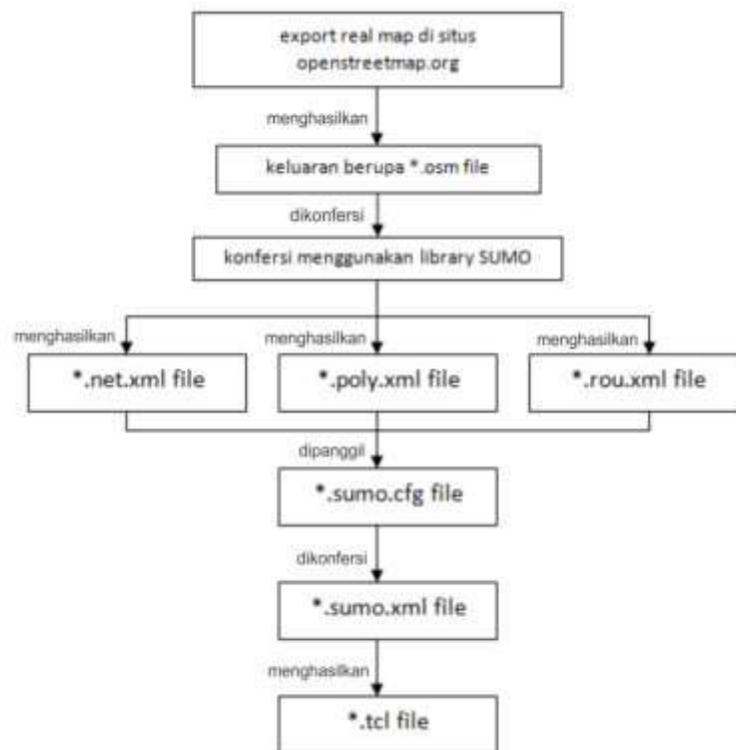
Tabel 3.3 Parameter Simulasi

Parameter	Nilai
Protokol <i>Routing</i>	TORA, MDART dan ZRP
Model Propagasi	<i>Free Space, TwoRayGround, Shadowing, dan Nakagami</i>
Protokol Pengiriman	UDP
Model Trafik	CBR
Jumlah <i>Node</i> di Jalan Kota Praya	35, 55 dan 70
Jumlah <i>Node di Bypass</i>	25, 40 dan 60
Kecepatan <i>Node</i> di Jalan Kota Praya	50 km/h, 60 km/h dan 80 km/h
Kecepatan <i>Node</i> di <i>Bypass</i>	60 km/h, 80 km/h dan 100 km/h
Pergerakan <i>Node</i>	<i>Random Waypoint</i>
Kanal	<i>Wireless</i>
Antena	<i>Omni Antenna</i>
MAC Layer	IEEE 802.11p

3.5 Perancangan Peta Simulasi

Perancangan peta simulasi diawali dengan mengambil area *real map* (studi kasus Jalan Kota Praya dan *Bypass* Lombok), agar dapat mengimplementasikan skenario dengan baik dan mengikuti *layout* dari ruas jalan yang ada pada lokasi yang dipilih penulis mengambil peta lokasi tersebut pada *openstreetmap*. *Openstreetmap* adalah situs pada internet yang menyediakan informasi peta dunia dimana setiap orang bias mengambil informasi tersebut dengan memasukkan titik koordinat dari suatu peta. Setelah mengunduh peta lokasi dari *openstreetmap*, penulis kemudian melakukan pengeditan untuk menyesuaikan lagi peta yang diunduh dengan hasil observasi yang telah dilakukan, pengeditan ini dilakukan dengan JOSM. JOSM dapat melakukan pengeditan seperti

penghapusan jalan gang yang tidak digunakan, bangunan dan lainnya yang tidak dibutuhkan dalam simulasi. Setelah peta *diedit*, selanjutnya buat sebuah *file type* yang mendefinisikan spesifikasi peta unduhan yang sudah *diedit*. Kemudian konversi dengan bantuan *netconvert* pada SUMO. Sehingga menghasilkan *file* baru yaitu *.net.xml, *.poly.xml, dan *.rou.xml yang dimana *file-file* tersebut berisi tentang informasi jumlah *node* yang bias ditentukan pada saat melakukan konversi, pergerakan dan rute *node* yang sesuai dengan *real map* (*.osm), serta informasi lainnya yang disesuaikan dengan bentuk *real map*. Kemudian *file* tersebut akan dipanggil melalui suatu *file* baru yaitu *.sumo.cfg yang sekaligus ditentukan berapa lama waktu simulasi. Kemudian *file* *.sumo.cfg akan dikonversi ke bentuk *.sumo.xml dan dikonversi lagi menjadi *.tcl *file*



Gambar 3.2 Diagram Blok Pembuatan Peta dengan SUMO

3.6 Perancangan Simulasi Jaringan VANET

Simulasi dilakukan dengan menggunakan *Network Simulator 2 (NS2)*, dimana untuk melakukan simulasi antara lain dibutuhkan sebuah pola trafik, skenario pergerakan *node* dan *script tcl*. Pada *.tcl *file* yang sebelumnya telah dibuat pada proses pembuatan peta, dan

kemudian akan dibuat sesuai dengan kebutuhan yaitu menggunakan tiga macam variasi protokol *routing* yakni protokol *routing* TORA, MDART dan ZRP. Kemudian dari masing-masing variasi protokol *routing* tersebut akan menggunakan empat macam variasi model propagasi yaitu model propagasi *freespace*, *tworayground*, *shadowing*, dan *nakagami*. Dan model propagasi tersebut akan menghasilkan keluaran berupa *trace file* (*.tr). Berikut adalah diagram alur proses simulasi pada penelitian ini seperti yang terdapat pada Gambar 3.3.



Gambar 3.3 Diagram Blok Simulasi

3.7 Parameter Uji atau *Quality of Service* (QoS)

Parameter uji atau biasa disebut dengan *Quality of Service* (QoS) diperlukan untuk mengetahui kualitas dari jaringan yang ada. Pengujian ini adalah untuk melihat performa dari protokol *routing* yang digunakan pada jaringan. Berikut adalah tabel parameter pengujian yang di gunakan dalam penelitian ini seperti pada Tabel 3.4.

Tabel 3.4 Parameter Pengujian Jaringan

No.	Parameter	Satuan
1	<i>Throughput</i>	Kbps
2	<i>Packet delivery ratio</i> (PDR)	%
3	<i>End-to-end delay</i> (E2D)	<i>Second</i>

1. *Throughput*

Throughput merupakan suatu istilah yang mendefinisikan banyaknya *bit* yang diterima dalam selang waktu tertentu dengan satuan kilo *bit per second* yang merupakan kondisi data *rate* sebenarnya dalam suatu jaringan.

2. *Packet delivery ratio* (PDR)

Packet delivery ratio merupakan perbandingan antara paket data yang berhasil diterima oleh *node* tujuan dengan total paket data yang dikirimkan oleh *node* sumber.

3. *End-to-end delay* (E2D)

Delay merupakan waktu pengiriman paket data dari sumber ke penerima, *delay* akan meningkat ketika kecepatan gerak *node* bertambah besar. Hal ini disebabkan pada saat kecepatan tinggi, maka kondisi jaringan akan semakin tidak stabil.

3.8 Jadwal Pelaksanaan Kegiatan

Rencana dan jadwal pelaksanaan penelitian tugas akhir ini direncanakan menghabiskan waktu selama 6 (enam) bulan yang diawali dengan studi literatur sampai dengan ujian tugas akhir 2. Rincian rencana pelaksanaan penelitian dapat dilihat pada Tabel 3.4.

Tabel 3.4 Rencana pelaksanaan Penelitian

No	Kegiatan	Bulan																					
		Mei-19	Jun-19	Jul-19	Agt-19	Sep-19	Okt-19	Nov-19	Des-19	Jan-20	Feb-20												
1	Studi Literatur	■	■	■	■	■	■	■	■														
2	Perancangan Simulasi			■	■	■	■	■	■														
3	Implementasi									■	■	■											
4	Uji Coba										■	■	■	■	■	■	■	■	■	■			
5	Analisa Hasil													■	■	■	■	■	■	■	■		
6	Kesimpulan																				■	■	
7	Penulisan Laporan																	■	■	■	■	■	■
8	Seminar TA 1								■	■													
9	Ujian TA 2																						■

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini dilakukan pembahasan dari skenario pengujian serta evaluasi hasil pengujian dengan melakukan analisa terhadap data hasil pengujian. Dari rencana penelitian yang telah di jelaskan pada bab sebelumnya, parameter yang digunakan adalah *throughput*, *packet delivery ratio* dan *end to end delay* yang disajikan dalam bentuk grafik.

4.1. Menentukan Skenario

Pada bagian ini dilakukan implementasi skenario penelitian dengan melakukan beberapa konfigurasi pada aplikasi JOSM, SUMO, dan NS-2. Berikut dijelaskan langkah-langkah yang dilakukan dalam penelitian ini.

4.1.1. Mengatur Parameter Trafik

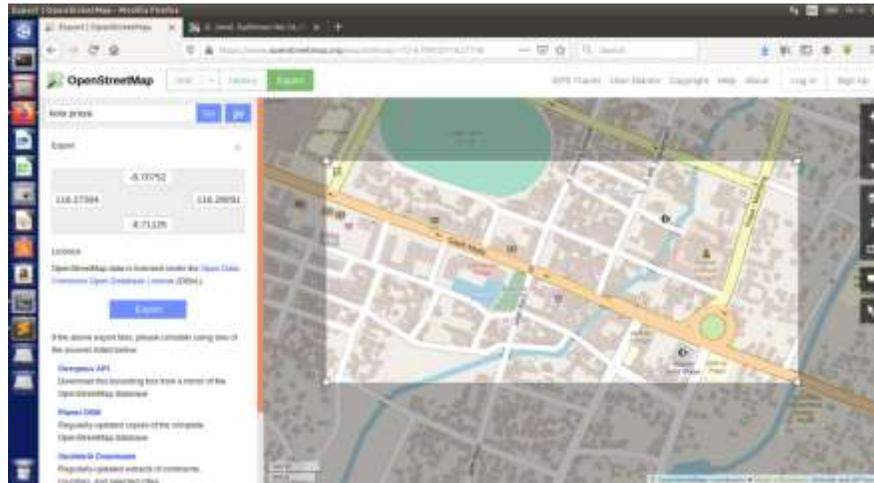
Pada bagian *script* ini, menggunakan beberapa parameter yang akan di tentukan untuk simulasi. Parameter yang digunakan yaitu tipe koneksi, jumlah *node* dalam simulasi, jumlah *seed*, maksimum koneksi yang dikehendaki, dan paket *rate*. Pengaturan ini dapat dilakukan pada direktori “*cmu-scen-gen*” yang terletak pada jalur direktori “*/ns-allinone-2.35/ns-2.35/indep-utils/*”. Perintah yang dijalankan pada terminal untuk melakukan pengaturan parameter trafik yang telah di konfigurasi seperti pada Tabel 4.1.

Tabel 4.1 Perintah pengaturan parameter trafik

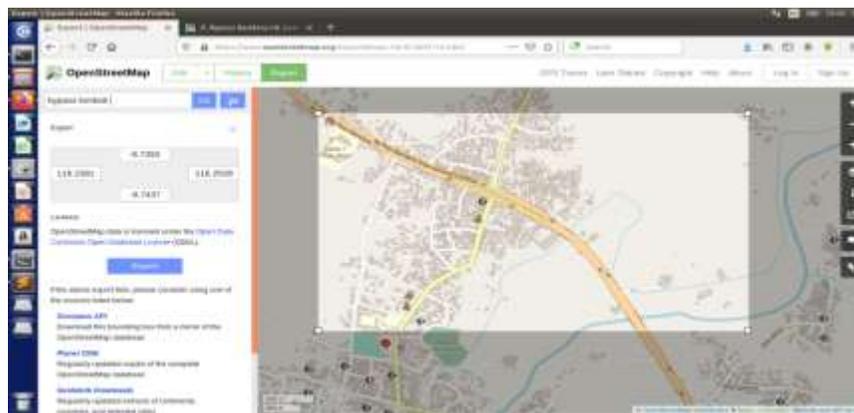
```
priyatama@priyatama:~/Downloads/ns-allinone-2.35/ns-2.35/indep-  
utils/cmu-scen-gen$ ns cbrgen.tcl -type cbr -nn 70 -seed 1 -mc 10 -  
rate 1.0 > cbrgen-node70-seed1.tcl
```

4.1.2. Mengunduh Peta

Pengunduhan peta dilakukan pada alamat <https://openstreetmap.org> dengan memasukkan nama daerah pada kolom *search* lalu klik *button* “*export*” (hijau) pada bagian atas, kemudian menyeleksi daerah yang akan digunakan, setelah itu klik *button* “*export*” pada bagian bawah (biru), maka *file.osm* akan muncul dan siap untuk diunduh. Tampilan saat mengunduh peta dapat dilihat pada Gambar 4.1 dan 4.2.



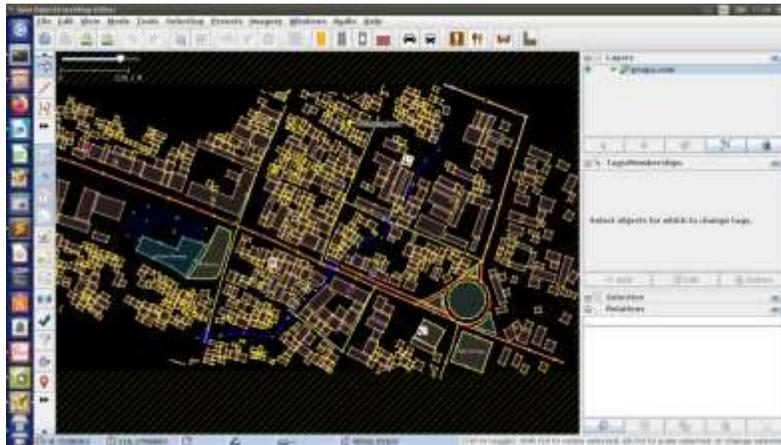
Gambar 4.1 Tampilan saat mengunduh peta Jalan Kota Praya



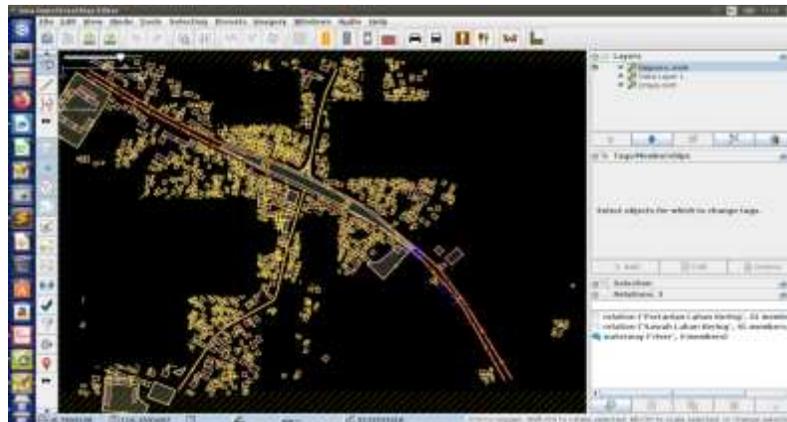
Gambar 4.2 Tampilan saat mengunduh peta *Bypass* Lombok

4.1.3. Mengedit Peta

JOSM adalah aplikasi yang digunakan untuk melakukan pengeditan *file*. osm yang telah diunduh sebelumnya. Pengeditan peta digunakan untuk merapikan dan membersihkan rute yang akan digunakan dari atribut gedung-gedung dan jalan buntu yang masih ada pada peta. Rute yang digunakan adalah rute jalan utama dalam peta yang ditandai dengan garis dengan berwarna merah pada peta jalan Kota Praya dan *Bypass* Lombok, dapat dilihat pada Gambar 4.3 dan 4.4.



Gambar 4.3 Peta Kota Praya yang sudah dirapikan



Gambar 4.4 Peta *Bypass* Lombok yang sudah dirapikan

4.1.4. Konversi *file* .OSM

Konversi *file* .osm dilakukan agar *file* peta yang telah diunduh sebelumnya dapat terbaca oleh aplikasi SUMO. Konversi *file* .osm ini terdiri dari beberapa langkah yaitu:

- Konversi dengan *netconvert*

Netconvert adalah program CLI yang pertama kali digunakan untuk mengubah *file* .osm menjadi *file* .net.xml. Perintah yang digunakan pada terminal dapat dilihat pada Tabel 4.2.

Tabel 4.2 Perintah konversi dengan *netconvert*

```
netconvert --osm-files file_name.osm --default.speed 28 -  
o file name.net.xml file name.net.xml
```

- Konversi dengan *randomtrips.py*

Randomtrips.py digunakan untuk memberikan rute acak yang akan dilalui oleh kendaraan. Perintah yang diberikan pada terminal dapat dilihat pada Tabel 4.3.

Tabel 4.3 Perintah konversi dengan *randomtrips.py*

```
python /usr/local/src/sumo-0.32.0/tools/randomTrips.py -n  
file name.net.xml -e 30 -o file name.trips.xml -s 4
```

- Konversi dengan *duarouter*

Pada langkah ini dilakukan penggabungan dua *file* hasil konversi dari perintah *netconvert* dan *randomtrips.py* sebelumnya dan memperbaiki *error* yang terjadi pada *file* lalu mengubahnya menjadi format *file* .rou.xml, *file* ini nantinya akan dipanggil oleh *file* .sumo.cfg bersamaan dengan *file* .net.xml hasil konversi dari *netconvert*. Perintah yang diberikan pada terminal dapat dilihat pada Tabel 4.4.

Tabel 4.4 Perintah konversi dengan *duarouter*

```
duarouter -n file_name.net.xml -r file_name.trips.xml --  
repair --ignore-errors -o file name.rou. file name.xml
```

4.1.5. Membuat *file* .sumo.cfg

File .sumo.cfg adalah *file* yang dapat dijalankan pada aplikasi SUMO, *file* ini berisikan *script* untuk menggabungkan dua *file* hasil konversi *file* .osm dan Panjang waktu simulasi yang digunakan. Membuat *file* ini dapat dilakukan dengan perintah “gedit nama *file* .sumo.cfg” pada terminal. Isi dari *file* .sumo.cfg dapat dilihat pada Tabel 4.5.

Tabel 4.5 Isi *file* .sumo.cfg

```
<configuration>  
<input>  
<net-file value=" file_name.net.xml"/>  
    <route-files value=" file_name.rou.xml"/>  
    <additional-files value=" file_name.poly.xml"/>  
</input>  
<time>  
    <begin value="0"/>  
    <end value="200"/>
```

```
<step-length value="0.1"/>
</time>
</configuration>
```

4.1.6. Membuat dan menjalankan *file* TCL

File TCL adalah *file* yang dapat dibaca oleh simulator NS-2. Sebelum mendapatkan *file* TCL, diperlukan konversi dari *file* .sumo.cfg yang telah didapatkan dari hasil konversi sebelumnya.

- Konversi *file* dengan SUMO

File .sumo.cfg adalah *file* yang dapat dibaca oleh simulator SUMO. Pada penelitian ini penulis menggunakan simulator SUMO untuk mengkonversi *file*.osm yang telah melalui beberapa konversi sebelumnya menjadi *file*.sumo.xml, *file* ini diperlukan untuk dikonversi kembali menjadi format lain agar dapat terbaca oleh NS-2. Perintah yang diberikan pada terminal dapat dilihat pada Tabel 4.6.

Tabel 4.6 Perintah konversi *file* dengan SUMO

```
sumo -c file_name.sumo.cfg --fcd-output file_name.sumo.xml
```

- Konversi dengan *traceexporter.py*

File output dari SUMO yang telah dilakukan dapat dikonversi menjadi *file* TCL. *File* TCL adalah *file* yang bias dibaca oleh simulator NS-2. Pada langkah ini penulis mengkonversi *file* .sumo.xml menjadi dua buah *file*, yakni *file map.tcl* yang berfungsi untuk memberikan pengaturan protokol yang digunakan dalam simulasi, dan sebuah *file* lagi yakni *file mobility.tcl* yang digunakan sebagai *file* konfigurasi pergerakan *node-node* dimana *node-node* akan bergerak secara *random* sesuai yang telah diatur pada tahap konversi dengan *randomtrips.py*. Perintah yang diberikan pada terminal dapat dilihat pada Tabel 4.7.

Tabel 4.7 Perintah konversi dengan *traceexporter.py*

```
python /usr/local/src/sumo-0.32.0/tools/traceExporter.py --
fcd-input file_name.sumo.xml --ns2config-output file_name.tcl
--ns2mobility-output mobility.tcl
```

`file_name.tcl` secara *default* berisi informasi parameter-parameter yang diberikan sebelumnya, parameter diberikan pada saat melakukan konversi pada *file.osm*. *File* TCL ini perlu diberikan modifikasi untuk dapat dijalankan oleh simulator NS-2. Isi dari *file* `file_name.tcl` dapat dilihat pada Tabel 4.8.

Tabel 4.8 Isi dari *file* TCL

```

set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation
model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue
type
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation
model
set val(netif) Phy/WirelessPhy ;# network interface
type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface
queue type
set val(ll) LL ;# link layer
type
set val(ant) Antenna/OmniAntenna ;# antenna
model
set val(ifqlen) 50 ;# max packet
in ifq
set val(nn) 25 ;# number of
mobilenodes
set val(rp) MDART ;# routing
protocol
set opt(x) 1068;
set opt(y) 861;

#=====

set ns_ [new Simulator]
set tracefd [open bypass.tr w]
$ns_ trace-all $tracefd

set namf [open bypass.nam w]
$ns_ namtrace-all-wireless $namf $opt(x) $opt(y)

# set up topography object
set topo [new Topography]

```

```

$topo load_flatgrid $opt(x) $opt(y)

#
# MAC Address IEEE 802.11p
#
source IEEE802-11p.tcl;

#=====

$ns_ node-config -adhocRouting $val(rp) \
                -llType $val(ll) \
                -macType $val(mac) \
                -ifqType $val(ifq) \
                -ifqLen $val(ifqlen) \
                -antType $val(ant) \
                -propType $val(prop) \
                -phyType $val(netif) \
                -channelType $val(chan) \
                -topoInstance $topo \
                -agentTrace ON \
                -routerTrace ON \
                -macTrace OFF \
                -movementTrace ON

    for {set i 0} {$i < $val(nn) } {incrit} {
        set node_($i) [$ns_ node]
        $node_($i) random-motion 0 ;# disable random motion
        $ns_ initial_node_pos $node_($i) 20
    }

source mobility.tcl
source cbrgen-node25-seed1.tcl
#=====

for {set i 0} {$i < $val(nn) } {incrit} {
    $ns_ at 300.0 "$node_($i) reset";
}
$ns_ at 300.0 "stop"
$ns_ at 300.01 "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd
    $ns_ flush-trace
    close $tracefd
}

puts "Starting Simulation..."
$ns_ run

```

Menjalankan *file* TCL dapat menggunakan perintah NS. Perintah yang digunakan pada terminal dapat dilihat pada Tabel 4.9.

Tabel 4.9. Perintah menjalankan *file* TCL

```
Ns file_name.tcl
```

4.1.7. Filtering Trace File Menggunakan Script AWK

Nilai QOS biasa didapatkan menggunakan *script* AWK, dengan *script* AWK ini akan dilakukan *filtering file trace* (.tr) hasil keluaran dari NS-2. Isi dari *file* AWK dapat dilihat pada Tabel 4.10.

Tabel 4.10 Isi *file* AWK

```
#####
#PACKET DELIVERY RATIO
#####
BEGIN {
sendLine = 0;
recvLine = 0;
fowardLine = 0;
}
$0 ~/^s.* AGT/ {
sendLine ++ ;
}
$0 ~/^r.* AGT/ {
recvLine ++ ;
}
$0 ~/^f.* RTR/ {
fowardLine ++ ;
}

END {
printf "\n > Packet Sent = %d",sendLine;
printf "\n > Packet Received = %d",recvLine;
printf "\n > Packet Delivery Ratio =
%.2f", (recvLine/sendLine)*100,"%";
printf "\n > Packet Loss = %.2f\n", ((sendLine-
recvLine)/sendLine)*100,"%";
}

#####
#TROUGHPUT
#####
BEGIN {
recvdSize = 0
```

```

startTime = 200
stopTime = 0
}
{
event = $1
time = $2
node_id = $3
pkt_size = $8
level = $4
# Store start time
if (level == "AGT"&& event == "s"&&pkt_size>= 512) {
if (time <startTime) {
startTime = time
}
}
# Update total received packets' size and store packets
arrival time
if (level == "AGT"&& event == "r"&&pkt_size>= 512) {
if (time >stopTime) {
stopTime = time
}
}
# Rip off the header
hdr_size = pkt_size % 512
pkt_size -= hdr_size
# Store received packet's size
recvdSize += pkt_size
}
}
END {
printf("\n > Average Throughput[kbps] =
%.2f\t\t", (recvdSize/(stopTime-startTime))*(8/1000))
}

#####
#END TO END DELAY
#####
BEGIN {
seqno = -1;
count = 0;
}
{
if($4 == "AGT"&& $1 == "s"&&seqno< $6) {
seqno = $6;
}
#end-to-end delay
if($4 == "AGT"&& $1 == "s") {
start_time[$6] = $2;
} else if(($7 == "tcp"||$7=="cbr") && ($1 == "r")) {
end_time[$6] = $2;
} else if($1 == "D"&& ($7 == "tcp"||$7=="cbr")) {

```

```

end_time[$6] = -1;
}
}
END {
for(i=0; i<=seqno; i++) {
if(end_time[i] > 0) {
delay[i] = end_time[i] - start_time[i];
count++;
}
else
{
delay[i] = -1;
}
}
for(i=0; i<=seqno; i++) {
if(delay[i] > 0) {
n_to_n_delay = n_to_n_delay + delay[i];
}
}
n_to_n_delay = n_to_n_delay/count;
print "\n > Average End-to-End Delay = " n_to_n_delay * 1000
" second";
print "\n";
}

```

Menjalankan proses *filtering file .tr* dapat dilakukan dengan perintah yang ada pada Tabel 4.11.

Tabel 4.11 perintah *filtering* dengan *file AWK*

```
awk -f file_name.awk file_name.tr
```

4.2. Hasil Uji Coba

Berikut ini merupakan hasil dari simulasi yang dilakukan pada NS-2 untuk protokol *routing* TORA (*Temporally Ordered Routing Algorithm*), MDART (*Multi-Path Dynamic Addressing Routing*) dan ZRP (*Zone Routing Protocol*). Parameter yang digunakan untuk menganalisis kinerja protokol *routing* adalah *throughput*, *average end-to-end delay* dan *packet delivery ratio (PDR)*. *File* keluaran dari simulator NS-2 dikelola menggunakan *script AWK*, dengan *script* ini didapatkan hasil pengujian dari parameter yang digunakan.

4.2.1. Analisis Nilai Rata-Rata Hasil Simulasi

Pada bagian ini disajikan hasil rata-rata dari simulasi yang telah dilakukan pada Jalan Kota Praya dan *Bypass* Lombok Hasil dapat dilihat pada Tabel 4.12 dan 4.13.

Tabel 4.12 Hasil simulasi di Jalan Kota Praya

	Propagasi	Parameter	Jumlah Node			Rata-rata
			35 node	55 node	70 node	
TORA	FreeSpace	Throughput	25,86	17,62	14,45	19,31
		PDR	95,86	86,25	82,25	88,12
		Delay	6,27	7,13	9,99	7,79
	TwoRayGround	Throughput	27,86	19,82	15,22	20,97
		PDR	96,25	88,63	83,93	89,60
		Delay	6,60	16,31	18,82	13,91
	Shadowing	Throughput	31,43	26,61	23,82	27,29
		PDR	97,48	83,82	64,88	82,06
		Delay	6,98	9,46	11,73	9,39
	Nakagami	Throughput	40,95	33,94	31,62	35,50
		PDR	98,55	95,76	92,83	95,71
		Delay	7,98	9,45	10,72	9,39
MDART	FreeSpace	Throughput	31,14	31,31	31,00	31,15
		PDR	99,14	99,02	99,07	99,08
		Delay	6,28	6,95	28,47	13,90
	TwoRayGround	Throughput	25,59	25,59	25,33	25,50
		PDR	81,76	80,99	81,17	81,31
		Delay	9,83	10,89	14,48	11,73
	Shadowing	Throughput	31,16	30,76	31,60	31,17
		PDR	99,34	98,41	99,35	99,03
		Delay	8,34	9,82	10,91	9,69
	Nakagami	Throughput	31,33	31,46	30,78	31,19
		PDR	99,02	98,83	99,06	98,97
		Delay	8,04	8,75	14,05	10,28
ZRP	FreeSpace	Throughput	31,52	26,81	21,11	26,48
		PDR	100,00	85,77	63,06	82,94
		Delay	34,47	2439,68	4844,86	2439,67
	TwoRayGround	Throughput	25,87	24,64	22,82	24,44
		PDR	81,45	78,17	55,75	71,79
		Delay	20,11	1021,43	2022,75	1021,43
	Shadowing	Throughput	31,60	26,01	22,45	26,69
		PDR	99,48	82,80	54,97	79,08
		Delay	40,04	2945,59	5851,14	2945,59
	Nakagami	Throughput	31,43	28,63	25,20	28,42
		PDR	99,02	90,90	64,44	84,79
		Delay	28,02	2206,12	7218,62	3150,92

Skenario uji coba yang pertama adalah di jalan Kota Praya dengan luas area 1000x1000 m² dengan hasil percobaan dilakukan dengan 35 node, 55 node dan 70 node. Dan menggunakan tiga protokol yaitu TORA, MDART dan ZRP.

1. Pada protokol TORA untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *nakagami* sebesar 35,50 Kbps, pada parameter PDR nilai

rata-rata pada propagasi *nakagami* sebesar 95,71% dan nilai rata-rata pada parameter *delay* sebesar 7,79 *second* pada propagasi *freespace*.

2. Pada protokol MDART untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *nakagami* sebesar 31,19 Kbps, pada parameter PDR nilai rata-rata pada propagasi *freespace* sebesar 99,08% dan nilai rata-rata pada parameter *delay* sebesar 9,69 *second* pada propagasi *shadowing*.
3. Pada protokol ZRP untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *nakagami* sebesar 28,42 Kbps, pada parameter PDR nilai rata-rata pada propagasi *nakagami* sebesar 84,79% dan nilai rata-rata pada parameter *delay* sebesar 1021,43 *second* pada propagasi *tworayground*.

Tabel 4.13 Hasil simulasi di jalan *Bypass* Lombok

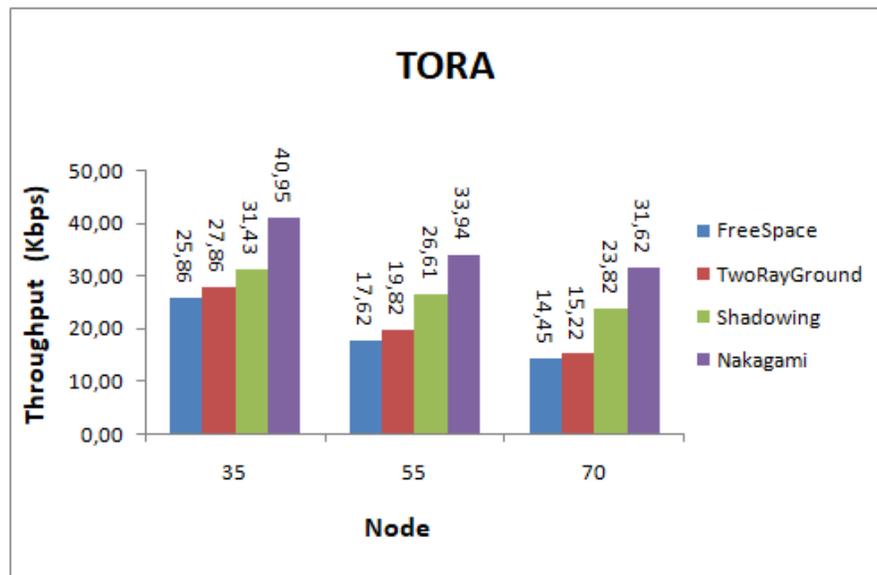
Protokol	Propagasi	Parameter	Jumlah Node			Rata-rata
			25 node	40 node	60 node	
TORA	FreeSpace	<i>Throughput</i>	20,35	15,82	11,92	16,03
		PDR	90,63	81,46	71,98	81,36
		<i>Delay</i>	4,82	6,63	10,57	7,34
	TwoRayGround	<i>Throughput</i>	23,22	16,45	12,56	17,41
		PDR	85,00	81,54	76,82	81,12
		<i>Delay</i>	6,99	14,79	16,44	12,74
	Shadowing	<i>Throughput</i>	28,29	23,56	19,23	23,69
		PDR	87,93	73,26	52,12	71,10
		<i>Delay</i>	7,00	8,46	10,83	8,76
	Nakagami	<i>Throughput</i>	30,63	23,44	21,93	25,33
		PDR	88,95	85,87	82,26	85,69
		<i>Delay</i>	6,93	9,92	11,46	9,44
MDART	FreeSpace	<i>Throughput</i>	31,29	31,11	31,12	31,17
		PDR	99,21	98,88	98,81	98,97
		<i>Delay</i>	6,20	7,12	7,98	7,10
	TwoRayGround	<i>Throughput</i>	25,64	19,81	22,50	22,65
		PDR	81,57	63,67	71,63	72,29
		<i>Delay</i>	15,98	42,67	49,67	36,11
	Shadowing	<i>Throughput</i>	31,45	31,45	31,57	31,49
		PDR	99,54	99,67	99,41	99,54
		<i>Delay</i>	8,07	10,60	11,40	10,02
	Nakagami	<i>Throughput</i>	30,94	31,14	31,56	31,21
		PDR	98,55	98,75	98,83	98,71
		<i>Delay</i>	7,98	9,45	10,72	9,39
ZRP	FreeSpace	<i>Throughput</i>	34,65	31,25	26,28	30,73
		PDR	99,84	99,80	83,73	94,46
		<i>Delay</i>	14,74	85,10	3186,12	1095,32
	TwoRayGround	<i>Throughput</i>	29,72	24,46	21,89	25,36
		PDR	85,91	70,03	69,59	75,18
		<i>Delay</i>	87,28	35,12	252,97	125,12
	Shadowing	<i>Throughput</i>	34,23	28,18	18,39	26,93
		PDR	99,68	90,70	58,50	82,96
		<i>Delay</i>	23,09	1470,22	3847,70	1780,34
	Nakagami	<i>Throughput</i>	34,50	34,18	29,30	32,66
		PDR	99,21	98,81	95,02	97,68
		<i>Delay</i>	12,06	29,18	618,62	219,96

Skenario uji coba yang kedua adalah di jalan *Bypass* Lombok dengan luas area 1000x1000 m² dengan hasil percobaan dilakukan dengan 25 *node*, 40 *node* dan 60 *node*. Dan menggunakan tiga protokol yaitu TORA, MDART dan ZRP.

4. Pada protokol TORA untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *nakagami* sebesar 25,33 Kbps, pada parameter PDR nilai rata-rata pada propagasi *nakagami* sebesar 85,69% dan nilai rata-rata pada parameter *delay* sebesar 7,34 *second* pada propagasi *freespace*.
5. Pada protokol MDART untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *shadowing* sebesar 31,49 Kbps, pada parameter PDR nilai rata-rata pada propagasi *shadowing* sebesar 99,54% dan nilai rata-rata pada parameter *delay* sebesar 7,10 *second* pada propagasi *freespace*.
6. Pada protokol ZRP untuk parameter *throughput* nilai rata-rata yang bagus didapatkan pada propagasi *nakagami* sebesar 32,66 Kbps, pada parameter PDR nilai rata-rata pada propagasi *nakagami* sebesar 97,68% dan nilai rata-rata pada parameter *delay* sebesar 125,12 *second* pada propagasi *tworayground*.

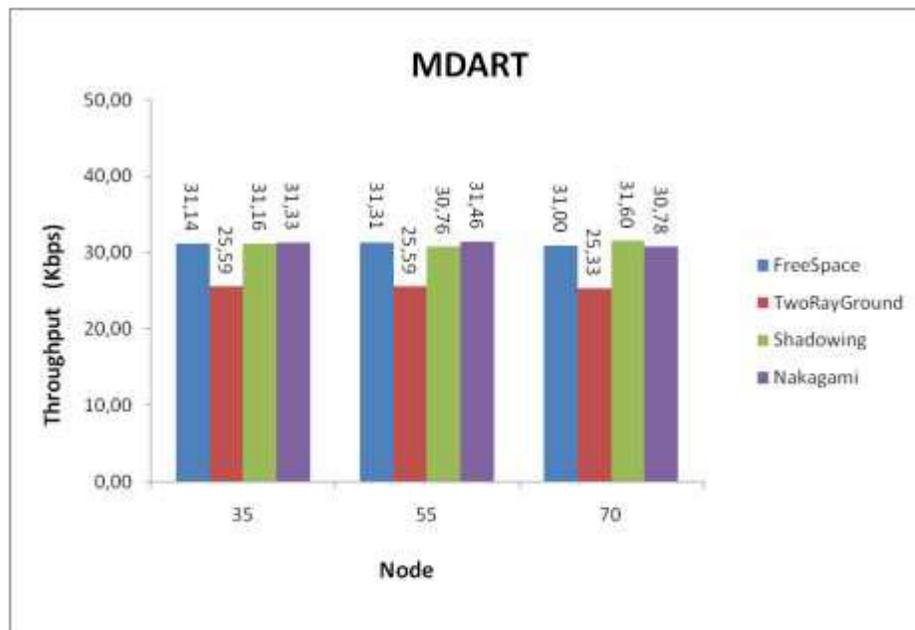
4.2.2. Hasil Simulasi Dijalan Kota Praya

4.2.2.1 Analisis *Throughput*



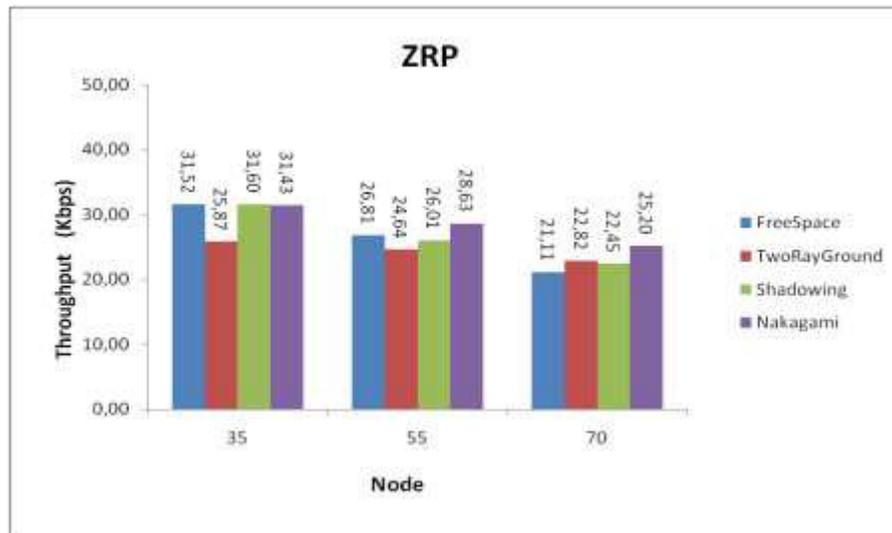
Gambar 4.5 *Throughput* protokol TORA

Pada Gambar 4.5 protokol TORA pada parameter *throughput* dengan luas area 1000x1000 m², terlihat bahwa nilai *throughput* yang paling tinggi pada propagasi *nakagami* dengan jumlah *node* 35 sebesar 40,95 Kbps. Sedangkan pada *node* 55 dan 70 nilai *throughput* terjadi penurunan dikarenakan seiring bertambahnya jumlah *node* nilai *throughput* kedua protokol ini menurun karena semakin tinggi jumlah kecepatan jarak antar *node* pun semakin menjauh. Dimana penyebabnya semakin tinggi kemungkinan terjadi *link failure* dan akibatnya nilai *throughput* menurun.



Gambar 4.6 *Throughput* protokol MDART

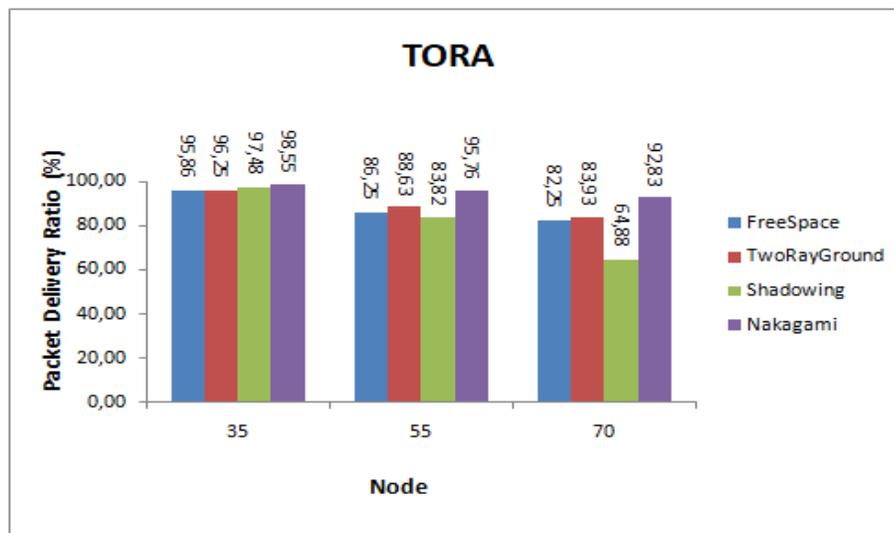
Pada gambar 4.6 protokol MDART dengan luas area 1000x1000 m², pada propagasi *freespace* bahwa nilai *throughput* yang paling tinggi pada propagasi *shadowing* dengan jumlah *node* 70 sebesar 31.60 Kbps. Dari keempat propagasi tersebut bahwa propagasi *tworayground* memberikan hasil yang kurang baik.



Gambar 4.7 *Throughput* protokol ZRP

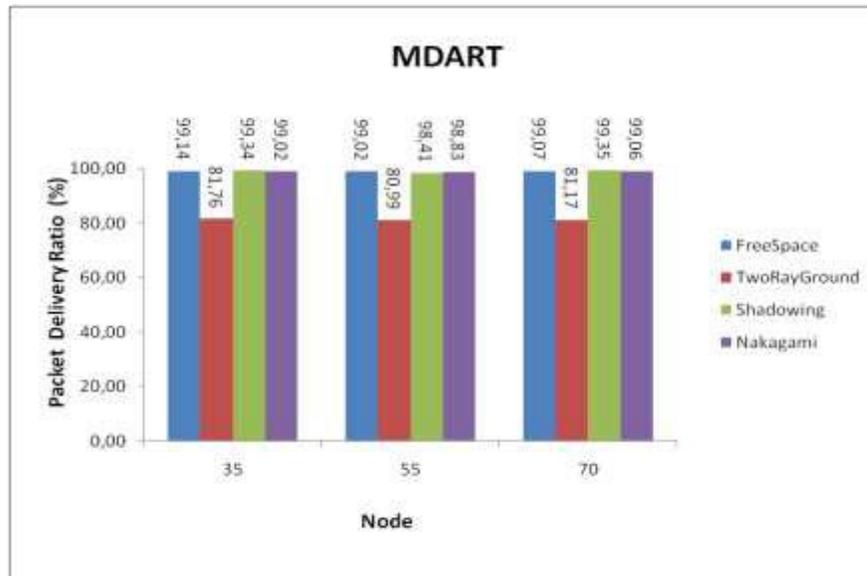
Pada Gambar 4.7 protokol ZRP dengan luas area 1000x1000 m², terlihat bahwa semakin *node* bertambah maka *throughput* yang dihasilkan untuk semua propagasi cenderung menurun. Nilai *throughput* yang paling tinggi pada propagasi *shadowing* dengan jumlah *node* 35 sebesar 31.60 Kbps sehingga memberikan nilai *throughput* yang lebih baik dari propagasi lainnya.

4.2.2.2 Analisis *Packet Delivery Ratio* (PDR)



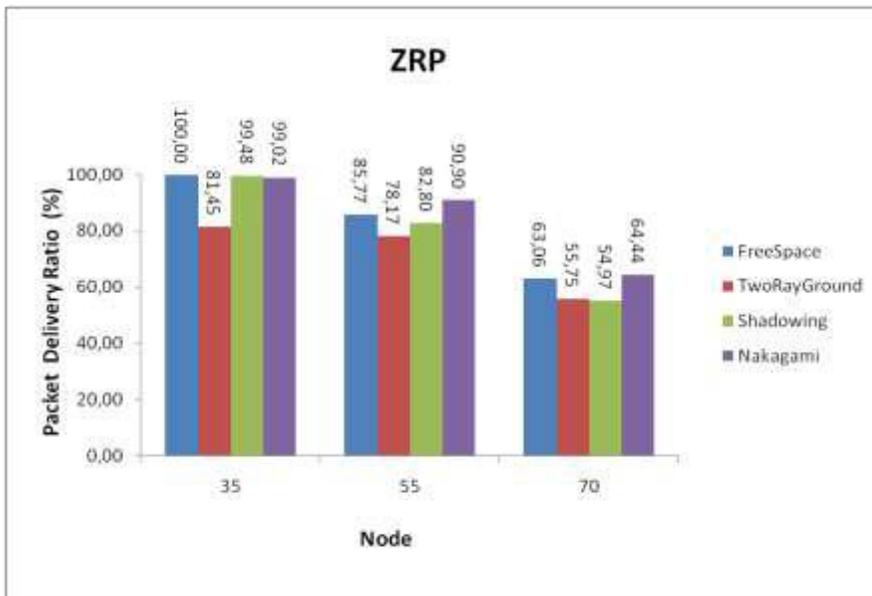
Gambar 4.8 *Packet delivery ratio* protokol TORA

Pada Gambar 4.8 protokol TORA dengan luas area 1000x1000 m² memiliki nilai yang dominan lebih tinggi. Protokol ini memiliki performa yang cukup stabil dalam parameter PDR. Ini disebabkan informasi yang selalu diberikan kepada *node* sumber apabila terjadi *link failure* akibat perubahan topologi dalam jaringan. Dari keempat propagasi tersebut bahwa propagasi *nakagami* memberikan nilai PDR yang lebih baik.



Gambar 4.9 Packet delivery ratio protokol MDART

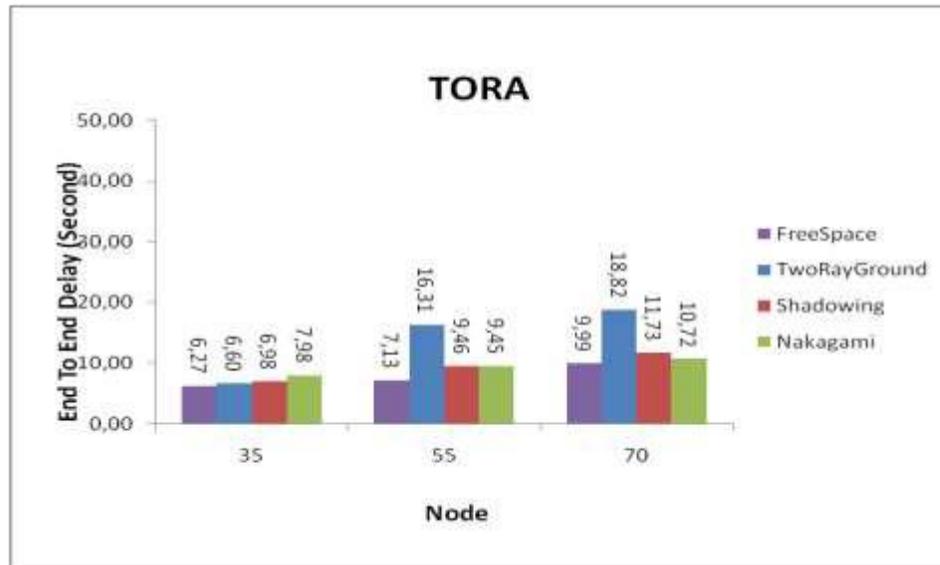
Protokol MDART dengan luas area 1000x1000 m², PDR merupakan presentase keberhasilan suatu data dikirim dengan semakin besar nilai PDR (mendekati nilai 100%) maka semakin baik kinerja suatu jaringan. Terlihat seperti di gambar 4.9 nilai PDR yang paling baik terletak di *node* 70 propagasi *shadowing* sebesar 99,35%, dari keempat propagasi tersebut bahwa propagasi *tworayground* memberikan hasil yang kurang baik. Ini dikarenakan pada protokol *routing* MDART kemungkinan untuk terjadi *link failure* semakin besar dan ketika *link failure* terjadi maka paket juga akan semakin banyak yang tertahan di *buffer*. Sehingga proses *update route* tersebut akan menyebabkan trafik jaringan menjadi padat sehingga berpengaruh terhadap keberhasilan pengiriman paket data.



Gambar 4.10 *Packet delivery ratio* protokol ZRP

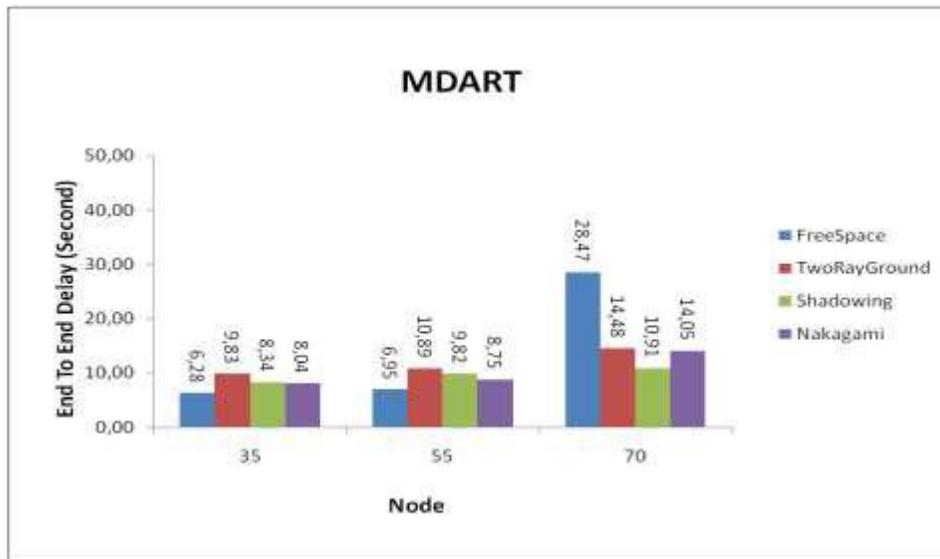
Pada Protokol ZRP dengan luas area 1000x1000 m², nilai PDR yang baik pada *node* 35 pada propagasi *freespace* mencapai nilai 100%. Presentase keberhasilan suatu data dikirim dengan semakin besar nilai PDR (mendekati nilai 100%) maka semakin baik kinerja suatu jaringan. Ketika kecepatan bertambah nilai PDR keempat propagasi tersebut mengalami penurunan, hal ini disebabkan karena banyaknya persimpangan di dalam skenario *urban* (Jalan Kota Praya) serta perpaduan kecepatan dan jumlah *node* didalam skenario, dimana terjadi banyak putusya jalur yang yang berakibat pada pengulangan pencarian rute baru.

4.2.2.3 Analisis *Average End-To-End Delay*



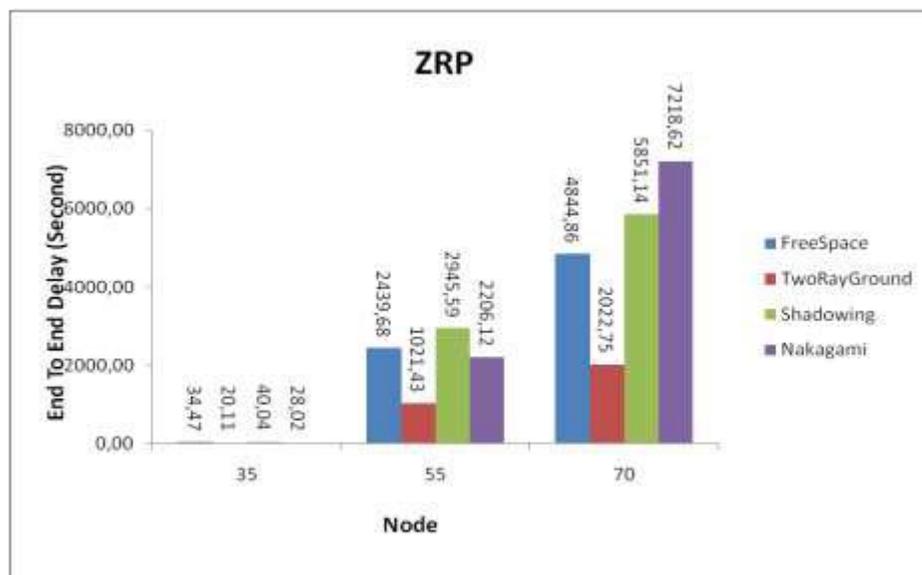
Gambar 4.11 *Delay* protokol TORA

Delay merupakan waktu tunda yang terjadi selama proses pengiriman data. Nilai kecil pada *delay* menunjukkan kinerja jaringan yang baik. Secara teori semakin banyak *node* maka nilai *delay* akan semakin besar dikarenakan waktu antrian paket dan waktu proses paket yang ditransmisikan.. Dilihat dari nilai rata-rata yang dihasilkan pada uji coba *end to end delay*, semakin banyak jumlah *node* yang digunakan maka *delay* yang dibutuhkan akan semakin meningkat. Hal ini disebabkan oleh *table routing node* akan di *update* apabila ada jalur yang terputus, sehingga menyebabkan *delay* terus meningkat ketika *node* semakin banyak dan kemungkinan *node* yang terputus akan semakin besar. Gambar 4.11 menunjukkan nilai *delay* paling baik pada *node* 35 pada propagasi *freespace* sebesar 6,27 *second*.



Gambar 4.12 Delay protokol MDART

Gambar 4.11 protokol MDART dengan luas area $1000 \times 1000 \text{m}^2$ menunjukkan nilai *delay* paling baik pada *node* 35 pada propagasi *freespace* sebesar 6,28 *second*. Karena semakin banyak *node* maka nilai *delay* akan semakin besar dikarenakan waktu antrian paket dan waktu proses paket yang ditransmisikan, seperti pada *node* 70 pada propagasi *freespace* dengan jumlah *delay* 28,47 *second* sehingga nilai *delay* nya kurang baik.

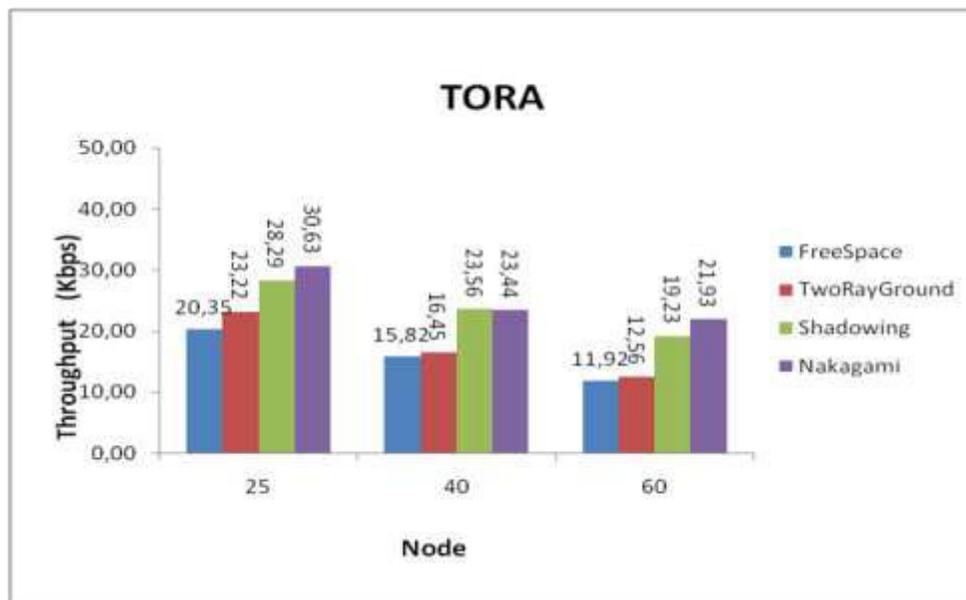


Gambar 4.13 Delay protokol ZRP

Pada Gambar 4.13 pada protokol ZRP dengan luas area $1000 \times 1000 \text{m}^2$ maka *delay* yang baik pada *node* 35 pada propagasi *tworayground* sebesar 20,11 *second*. Dilihat dari nilai rata-rata yang dihasilkan pada uji coba *end to end delay*, semakin banyak jumlah *node* yang digunakan maka *delay* yang dibutuhkan akan semakin meningkat. Hal ini disebabkan oleh *table routing node* akan di *update* apabila ada jalur yang terputus, sehingga menyebabkan *delay* terus meningkat ketika *node* semakin banyak dan kemungkinan *node* yang terputus akan semakin besar.

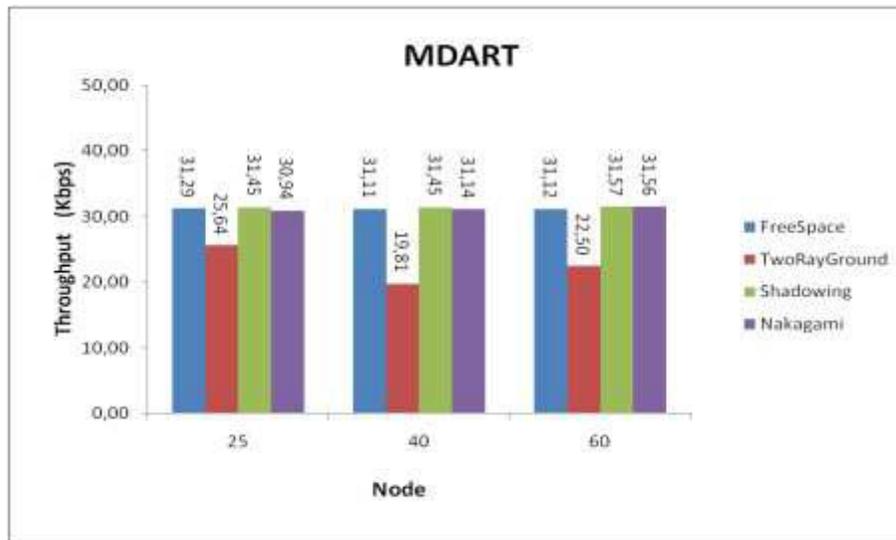
4.2.3 Hasil Simulasi Di Bypass Lombok

4.2.3.1 Analisis *Throughput*



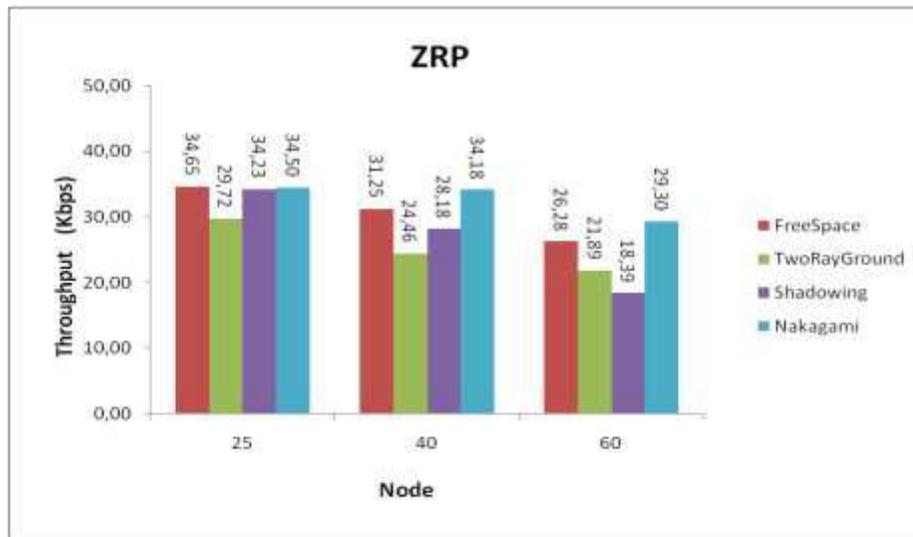
Gambar 4.14 *Throughput* protokol TORA

Pada Gambar 4.14 protokol TORA pada parameter *throughput* dengan luas area $1000 \times 1000 \text{m}^2$, terlihat bahwa nilai *throughput* yang paling tinggi pada propagasi *nakagami* dengan jumlah *node* 25 sebesar 30,63 Kbps. Sedangkan pada *node* 40 dan 60 nilai *throughput* terjadi penurunan dikarenakan seiring bertambahnya kecepatan *node* nilai *throughput* kedua protokol ini menurun karena semakin tinggi jumlah kecepatan jarak antar *node* pun semakin menjauh. Dimana penyebabnya semakin tinggi kemungkinan terjadi *link failure* dan akibatnya nilai *throughput* menurun.



Gambar 4.15 *Throughput* protokol MDART

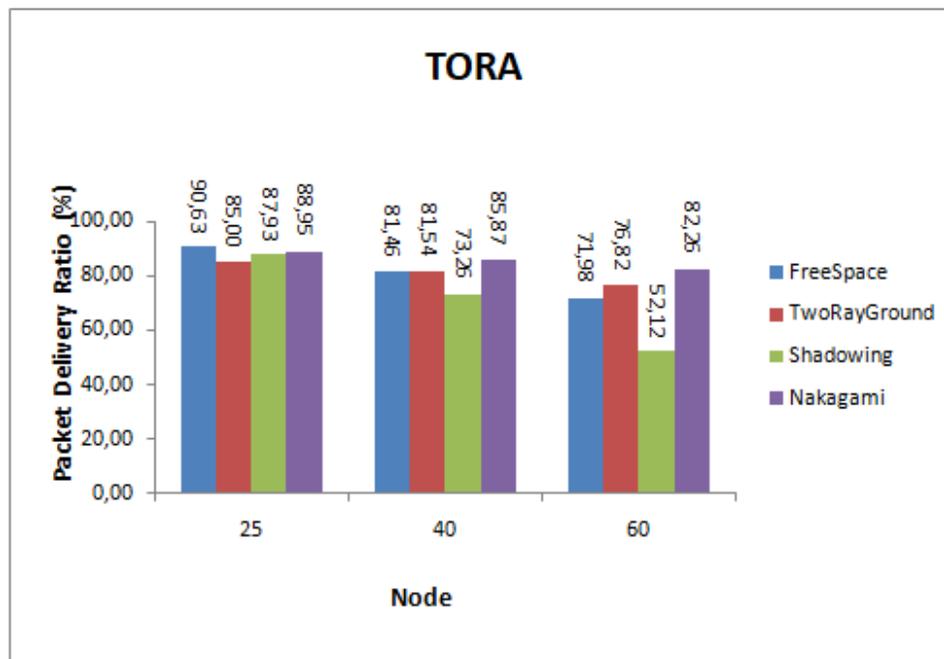
Nilai *throughput* dipengaruhi oleh beberapa hal, dimana salah satunya adalah konsistensi dari topologi jaringan tersebut dan proses pencarian jalur tersebut sehingga jalur tersebut terbentuk. Pada gambar 4.15 protokol MDART dengan luas area 1000x1000 m², bahwa nilai *throughput* yang paling tinggi pada propagasi *nakagami* dengan jumlah *node* 60 sebesar 31.56 Kbps. Dari keempat propagasi tersebut bahwa propagasi *tworayground* memberikan hasil yang kurang baik.



Gambar 4.16 *Throughput* protokol ZRP

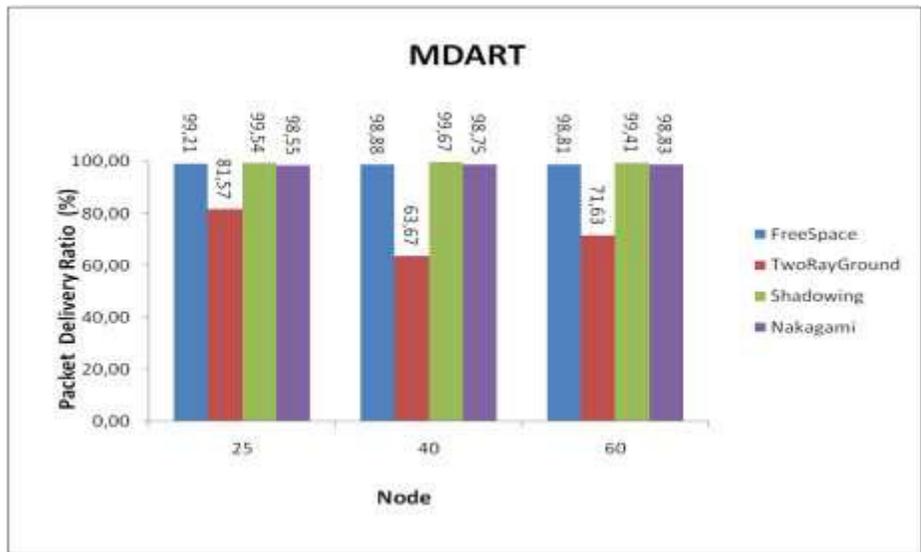
Pada Gambar 4.16 protokol ZRP dengan luas area 1000x1000 m², terlihat bahwa semakin *node* bertambah maka *throughput* yang dihasilkan untuk semua propagasi cenderung menurun. Nilai *throughput* yang paling tinggi pada propagasi *freespace* dengan jumlah *node* 25 sebesar 34.65 Kbps sehingga memberikan nilai *throughput* yang lebih baik dari propagasi lainnya.

4.2.3.2 Analisis *Packet Delivery Ratio* (PDR)



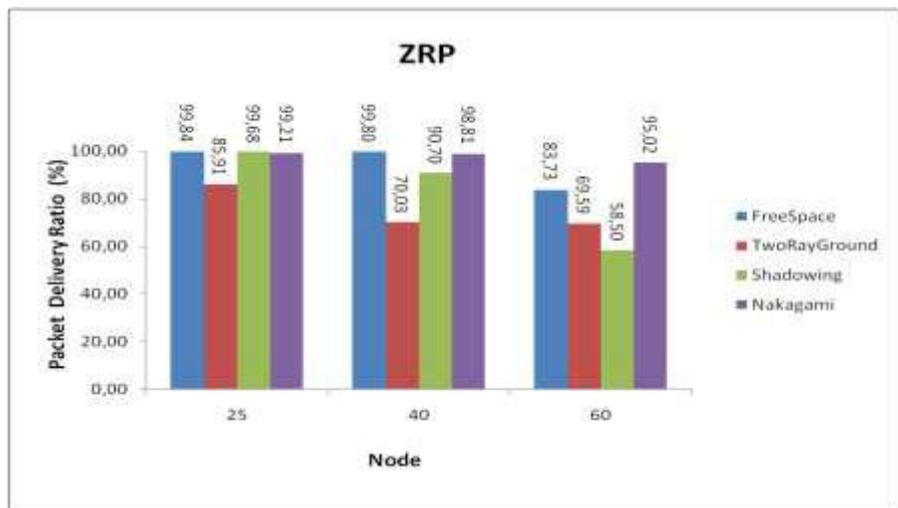
Gambar 4.17 *Packet delivery ratio* protokol TORA

PDR dipengaruhi oleh nilai *throughput* semakin tinggi nilai PDR suatu *routing protocol* semakin bagus juga *routing protocol* tersebut dalam menentukan atau mencari rute. Pada hasil rata-rata pada skenario di *Bypass* ini protokol TORA mengalami penurunan nilai PDR. Gambar 4.17 pada *node* 25 propagasi *freespace* senilai 90,63% memiliki nilai yang dominan lebih tinggi dari propagasi lainnya. protokol ini memiliki performa yang cukup stabil dalam parameter PDR. Ini disebabkan informasi yang selalu diberikan kepada *node* sumber apabila terjadi *link failure* akibat perubahan topologi dalam jaringan selalu tersedia sehingga TORA yang mendukung *multi route* dapat memilih jalur lain untuk mengirimkan pakatnya.



Gambar 4.18 *Packet delivery ratio* protokol MDART

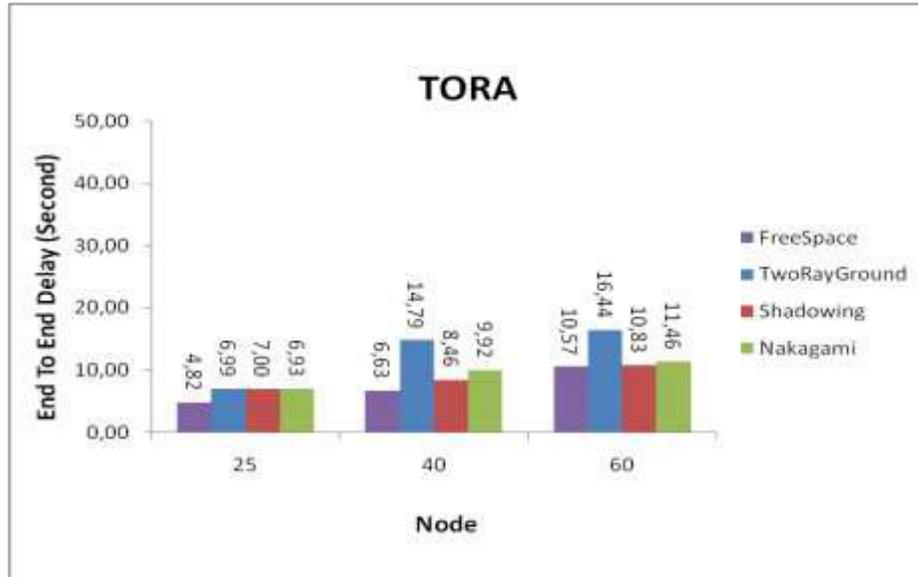
PDR dipengaruhi oleh nilai *throughput* semakin tinggi nilai PDR suatu *routing protocol* semakin bagus juga *routing protocol* tersebut dalam menentukan atau mencari rute. Protokol MDART dengan luas area 1000x1000 m², seperti pada gambar 4.18 nilai PDR yang paling baik terletak di *node* 40 propagasi *shadowing* sebesar 99,67%, semakin tinggi nilai PDR semakin bagus pula *routing protocol* tersebut. Dari keempat propagasi tersebut bahwa propagasi *tworayground* memberikan hasil yang kurang baik.



Gambar 4.19 *Packet delivery ratio* protokol ZRP

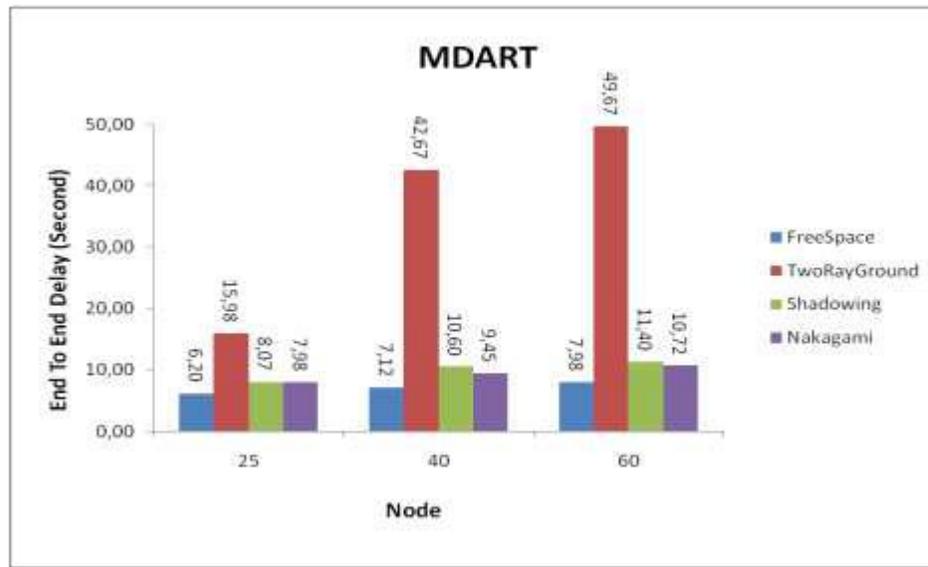
Pada Gambar 4.19 protokol ZRP dengan luas area 1000x1000 m², PDR merupakan presentase keberhasilan suatu data dikirim dengan semakin besar nilai PDR (mendekati nilai 100%) maka semakin baik kinerja suatu jaringan. sehingga pada protokol ZRP nilai PDR yang baik pada *node 25* pada propagasi *freespace* mencapai nilai 99.84%.

4.2.3.3 Analisis *Average End-To-End Delay*



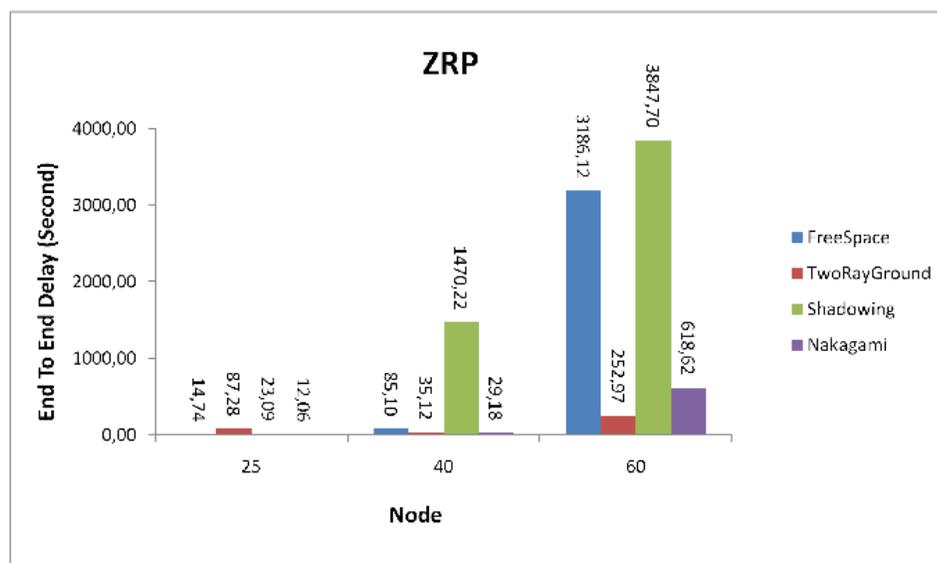
Gambar 4.20 *Delay* protokol TORA

Delay merupakan waktu tunda yang terjadi selama proses pengiriman data. Nilai kecil pada *delay* menunjukkan kinerja jaringan yang baik. Secara teori semakin banyak *node* maka nilai *delay* akan semakin besar dikarenakan waktu antrian paket dan waktu proses paket yang ditransmisikan. Gambar 4.20 menunjukkan nilai *delay* paling baik pada *node 25* pada propagasi *freespace* sebesar 4,82 *second*.



Gambar 4.21 Delay protokol MDART

Gambar 4.21 protokol MDART dengan luas area 1000x1000m² menunjukkan nilai *delay* paling baik pada *node* 25 pada propagasi *freespace* sebesar 6,20 *second*. Karena semakin banyak *node* maka nilai *delay* akan semakin besar dikarenakan waktu antrian paket dan waktu proses paket yang ditransmisikan, seperti pada *node* 60 pada propagasi *tworayground* dengan jumlah *delay* 49,67 *second* sehingga nilai *delay* nya kurang baik.



Gambar 4.22 Delay protokol ZRP

Pada Gambar 4.22 nilai *end to end delay* untuk protokol ZRP lebih tinggi. dikarenakan pergerakan *node* yang *random* dilihat dari nilai rata-rata yang dihasilkan pada uji coba *end to end delay*, semakin banyak jumlah *node* yang digunakan maka *delay* yang dibutuhkan akan semakin meningkat dan mempengaruhi kemungkinan terjadinya topologi atau kegagalan *route* atau *link failure* semakin besar. Dan luas area yang lebih kecil topologi yang terjadi menjadi lebih padat, sehingga pada protokol ZRP yang masuk dalam jenis protokol *hybrid* banjir dengan informasi *update* tabel *routing* membuat pengiriman paket dari *node* sumber ke *node* tujuan mengalami penundaan yang cukup lama.

BAB V

KESIMPULAN DAN SARAN

5.1. KESIMPULAN

Setelah dilakukan analisis dari hasil simulasi jaringan VANET yang menggunakan *routing* protokol TORA, MDART dan ZRP, didapatkan kesimpulan sebagai berikut ini :

1. Pada skenario urban atau di peta jalan Kota Praya dengan luas area 1000x1000 m², nilai rata-rata yang terbaik pada protokol TORA parameter *throughput* sebesar 35,50 Kbps pada propagasi *nakagami* dan *delay* sebesar 7,79 *second* pada propagasi *freespace*, untuk nilai rata-rata parameter PDR diperoleh sebesar 99,08% pada protokol MDART propagasi *freespace*.
2. Pada skenario *rural* atau *Bypass* Lombok parameter *throughput* hasil nilai rata-rata yang terbaik didapatkan pada protokol ZRP 32,66 Kbps pada propagasi *nakagami*. Pada protokol MDART parameter PDR nilai rata-rata sebesar 99,54% pada propagasi *shadowing* dan untuk parameter *delay* sebesar 7,10 *second* pada propagasi *freespace*.
3. Dari hasil percobaan pada skenario urban di jalan peta Kota Praya protokol yang cocok untuk digunakan yaitu protokol TORA dan di sisi rural protokol yang cocok digunakan yaitu protokol MDART, serta dari kedua skenario tersebut propagasi yang bagus untuk digunakan yaitu propagasi *nakagami*.

5.2. SARAN

Berdasarkan kesimpulan yang sudah dipaparkan, maka dapat diusulkan beberapa saran berikut demi pengembangan kedepannya, seperti :

1. Melakukan penelitian serupa dengan menambahkan beberapa algoritma yang bisa digunakan serta menggunakan NS3 agar VANET benar-benar bisa di implementasikan kedepannya.
2. Melakukan penelitian selain protokol TORA, MDART dan ZRP pada jaringan VANET.

DAFTAR PUSTAKA

- [1] S. Ramadhani, and R. Anggoro, “Studi perbandingan kinerja model transmisi TwoRayGround dan Nakagami pada OLSR di,” *J. Teknik ITS*, vol. 5, no. 2, 2016.
- [2] A. K. Gupta, H. Sadawarti, and A. K. Verma, “Performance analysis of AODV , DSR & TORA Routing Protocols,” vol. 2, no. 2, 2010. *Int. J. Engineering and Tecknology*, vol. 2. no. 2, 2010.
- [3] V. Jain and R. S. Kushwah, “Review of various VANET protocols using NS-2 simulator,” *Int. J. Comput.*, vol. 4, no. 7, 2016.
- [4] A. Ardiansyah, D. Perdana, and R. F. Sari, “Performance comparison of GPSR and ZRP routing protocols in VANET environment,” IEEE region 10 TENSymp, 2016.
- [5] R. C. Poonia and V. Singh, “performance evaluation of radio propagation model for vehicular adhoc networks using vanetmobisim and NS-2,” *Int. J. Distributed and Parallel Systems*, vol. 3, no. 4, hal 145–155, 2012.
- [6] B. Makodia, D. Marsona, “Effect of radio-wave propagation model on network performance with AODV routing protocol for IL-VANET,” *Int. Arch.App. Scl. Technol*, vol 5, hal 13–18, 2014.
- [7] K. A. Hafeez, L. Zhao, Z. Liao, and B. N. Ma, “The optimal radio propagation model in VANET The optimal radio propagation model in VANET,” *Int. J. Elektrical and Compt. Engineering*, 2015.
- [8] P. D. Setiady, F. A. Yulianto, and A. Herutomo, “Analisis perbandingan performansi routing protocol MDART dan ZRP pada jaringan vehicular ad hoc network,” *J. Fak. Teknik Informatika, Univ. Telkom*, 2014.
- [9] E. Haryatmi, B. Soerowirdjo, and A. B. Mutiara, “Pengaruh propagasi terhadap komunikasi data pada jaringan nirkabel,” *ISBN: 979-756-061-6*, 2005.
- [10] M. D. Agustya, “Analisis performa proactive routing protocol DSDV dan OLSR pada vehicular adhoc network (VANET) menggunakan NS-3,” skripsi, Sepuluh Nopember Intitute of Technology, 2015.
- [11] P. K. Singh, “Influences of TwoRayGround and Nakagami propagation model for the performance of adhoc routing protocol in VANET,” *Int. J. comput. applications*, vol. 45, no. 22, pp. 1–6, 2012.

