

**DAMPAK VARIASI MODEL PROPAGASI TERHADAP PROTOKOL  
*ROUTING* REAKTIF, PROAKTIF DAN *HYBRID* DI JARINGAN VANET  
(STUDI KASUS KOTA BIMA)**

Tugas Akhir  
Untuk memenuhi sebagai persyaratan  
mencapai derajat Sarjana S-1 Program studi Teknik Informatika



Oleh:

**Nazibullah  
F1D 013 073**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS MATARAM**

**2020**

## TUGAS AKHIR

### DAMPAK VARIASI MODEL PROPAGASI TERHADAP PROTOKOL ROUTING REAKTIF, PROAKTIF DAN HYBRID DI JARINGAN VANET (STUDI KASUS KOTA BIMA)

Oleh :

**NAZIBULLAH**

F1D013073

Telah diperiksa oleh Tim Pembimbing :

1. Pembimbing Utama



Tanggal: 26/07/2020

**Andy Hidayat Jatmika, S.T., M.Kom.**  
**NIP. 198312092012121001**

2. Pembimbing Pendamping



Tanggal: 26/07/2020

**Fitri Bimantoro, ST., M.Kom.**  
**NIP. 198606222015041002**

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Teknik  
Universitas Mataram



**Prof. Dr. Eng. I Gede Pasek Suta Wijaya, ST., MT.**  
**NIP. 197311302000031001**

## TUGAS AKHIR

### DAMPAK VARIASI MODEL PROPAGASI TERHADAP PROTOKOL ROUTING REAKTIF, PROAKTIF DAN HYBRID DI JARINGAN VANET (STUDI KASUS KOTA BIMA)

Oleh :

**NAZIBULLAH**  
F1D013073

Telah diujikan di depan penguji  
Pada tanggal 13 Juli 2020  
Dan dinyatakan telah memenuhi syarat mencapai derajat Sarjana S-1  
Program Studi Teknik Informatika

Susunan Tim Penguji :

1 Penguji 1



Tanggal: 24/07/2020

**Nadiyahari Agitha, S.Kom., M.MT**  
**NIP. 198608132018032001**

2 Penguji 2



Tanggal: 23/07/2020

**Gibran Satya Nugraha, S.Kom., M.Eng.**  
**NIP. 199203232019031012**

3 Penguji 3



Tanggal: 25/07/2020

**Arik Aranta, S.Kom., M.Kom.**  
**NIP. 199402202019031004**

Mataram,  
Dekan Fakultas Teknik  
Universitas Mataram



**Akmaluddin, ST., M.Sc Eng., Ph.D.**  
**NIP. 196812311994121001**

## **HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR**

Saya yang bertanda tangan di bawah ini menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi 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

Mataram, Juli 2019

Nazibullah

## **PRAKATA**

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Pengasih dan Penyayang atas segala berkat, bimbingan, dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan Tugas Akhir dengan judul “*Dampak Variasi Model Propagasi Terhadap Protokol Routing Reaktif, Proaktif Dan Hybrid Di Jaringan VANET (Studi Kasus Kota Bima)*”.

Tugas Akhir ini dilaksanakan di Laboratorium Embedded System and Data Communications, Program Studi Teknik Informatika UNRAM. Tujuan dari Tugas Akhir ini adalah untuk memberikan gambaran bagaimana dampak berbagai variasi model propagasi terhadap protokol routing reaktif, proaktif, dan hybrid pada jaringan VANET pada peta jalan Kota Bima. Tugas Akhir ini juga merupakan salah satu persyaratan kelulusan guna mencapai gelar kesarjanaan di Program Studi Teknik Informatika, Fakultas Teknik UNRAM.

Akhir kata semoga tidaklah terlampau berlebihan, bila penulis berharap agar karya ini dapat bermanfaat bagi pembaca.

Mataram, Juli 2019

Penulis

## UCAPAN TERIMAKASIH

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

1. Kedua orang tua saya, dan seluruh keluarga yang telah mendukung dan mendoakan.
2. Bapak Akmaluddin, ST., M.Sc.(Eng.), Ph.D., selaku Dekan Fakultas Teknik Universitas Mataram.
3. Bapak Prof. Dr. Eng. I Gede Pasek Suta Wijaya, ST., MT., selaku Ketua Program Studi Teknik Informatika Fakultas Teknik Universitas Mataram.
4. Bapak Andy Hidayat Jatmika, ST., M. Kom., selaku Dosen Pembimbing Utama yang telah memberikan bimbingan, arahan, serta motivasi kepada penulis selama mengerjakan Tugas Akhir.
5. Bapak Fitri Bimantoro, ST.,M.Kom., selaku Dosen Pembimbing Pendamping yang telah turut memberikan ide, saran, arahan, serta motivasi kepada penulis dalam mengerjakan Tugas Akhir.
6. Ibu Nadiyahari Agitha, S.Kom., M.Kom. selaku Dosen Penguji 1, Bapak Gibran Satya Nugraha, S.Kom., M.Eng. selaku Dosen Penguji 2, Bapak Arik Aranta, S.Kom., M.Kom. selaku Dosen Penguji 3, yang telah memberikan saran dan arahan kepada penulis dalam mengerjakan Tugas Akhir.
7. Kepada saudara Mery, Gifari, Bima, Jeki, Badrun yang turut mendukung dan membantu.
8. Teman-teman seperjuangan Teknik Informatika angkatan 2013, yang turut mendukung dan membantu.
9. Semua pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan bimbingan dan dukungan kepada penulis dalam menyelesaikan Tugas Akhir ini.

## DAFTAR ISI

COVER

LEMBAR PENGESAHAN.....	ii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
PRAKATA .....	v
UCAPAN TERIMAKASIH.....	vi
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL.....	xiii
ABSTRAK.....	xiv
ABSTRACT.....	xv
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
<b>BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....</b>	<b>4</b>
2.1 Tinjauan Pustaka.....	4
2.2 Dasar Teori .....	7
2.2.1 Jaringan Komputer.....	7
2.2.2 Jaringan Nirkabel <i>Ad-Hoc (Ad-Hoc Wireless Network)</i> .....	7
2.2.3 Vehicular Ad-Hoc Network (VANET).....	8
2.2.4 Protokol <i>Routing</i> .....	8
2.2.5 Protokol <i>Routing</i> Reaktif.....	9

2.2.6	Protokol <i>Routing</i> Proaktif.....	9
2.2.7	Protokol <i>Routing Hybrid</i> .....	9
2.2.8	Protokol <i>Routing Ad-Hoc on Demand Multipath Distance Vector</i> (AOMDV).....	9
2.2.9	Protokol <i>Routing Optimized Link State Routing</i> (OLSR) .....	11
2.2.10	Protokol <i>Routing Zone Routing Protocol</i> (ZRP).....	11
2.2.11	Propagasi Gelombang Radio.....	12
2.2.12	Model Propagasi <i>Free Space</i> .....	13
2.2.13	Model Propagasi <i>TwoRayGround</i> .....	13
2.2.14	Model Propagasi <i>Shadowing</i> .....	13
2.2.15	Model Propagasi <i>Nakagami</i> .....	15
2.2.16	Parameter <i>Quality of Service</i> .....	15
2.2.17	<i>Throughput</i> .....	16
2.2.18	<i>Packet Loss</i> .....	16
2.2.19	<i>End to End Delay</i> (E2D).....	16
2.2.20	<i>Network Simulator 2</i> (NS2).....	16
2.2.21	<i>Simulation of Urban Mobility</i> (SUMO).....	17
2.2.22	<i>Java OpenStreetMap</i> (JOSM).....	17
2.2.23	<i>AWK Script</i> .....	18
<b>BAB III METODE PENELITIAN .....</b>		<b>19</b>
3.1	Identifikasi Kebutuhan Sistem .....	19
3.1.1	Kebutuhan Perangkat Keras.....	19
3.1.2	Kebutuhan Perangkat Lunak.....	19
3.2	Diagram Alir Penelitian.....	19
3.3	Perancangan Skenario Simulasi .....	20
3.3.1	Parameter Simulasi.....	21
3.3.2	Perancangan Peta Simulasi.....	21
3.3.3	Perancangan Simulasi Dengan NS2 .....	22
3.4	Pengujian.....	23
3.5	Jadwal Pelaksanaan Kegiatan.....	24

<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>25</b>
4.1 Merancang Skenario Simulasi.....	25
4.1.1 Membuat Peta Simulasi.....	25
4.1.2 Membuat Rute Berdasarkan Peta ( <i>osm file</i> ).....	26
4.1.3 Pembuatan <i>tcl file</i> .....	27
4.2 Menjalankan Simulasi.....	32
4.3 Analisis Hasil Menggunakan AWK.....	33
4.4 Hasil Uji Coba.....	33
4.4.1 Unjuk Kinerja Berdasarkan Nilai <i>Packet Loss</i> .....	33
4.4.2 Unjuk Kinerja Berdasarkan Nilai <i>Throughput</i> .....	34
4.4.3 Unjuk Kinerja Berdasarkan Nilai <i>End To End Delay</i> .....	35
4.5 Analisis Hasil Uji Coba.....	35
 <b>BAB V KESIMPULAN DAN SARAN.....</b>	 <b>40</b>
5.1 Kesimpulan.....	40
5.2 Saran.....	40
 <b>DAFTAR PUSTAKA .....</b>	 <b>41</b>

## DAFTAR GAMBAR

Gambar 2.1 Ilustrasi Topologi AOMDV.....	10
Gambar 2.2 Ilustrasi Topologi OLSR .....	11
Gambar 2.3 Ilustrasi Topologi ZRP .....	12
Gambar 2.4 Tampilan GUI NS2 .....	17
Gambar 2.5 Tampilan GUI SUMO .....	17
Gambar 3.1 Diagram Alir Penelitian.....	20
Gambar 3.2 Diagram Alir Pembuatan Peta Dengan SUMO .....	22
Gambar 3.3 Diagram Alir Simulasi.....	23
Gambar 3.4 Tahap Pengujian .....	24
Gambar 4.1 Situs Openstreetmap.org .....	25
Gambar 4.2 Peta disunting menggunakan <i>software</i> JOSM.....	26
Gambar 4.3 Diagram alir pembuatan <i>file tcl</i> .....	28
Gambar 4.4 Animasi simulasi dengan NAM.....	32
Gambar 4.5 Perbandingan nilai rata-rata <i>packet loss</i> . (a) grafik perbandingan nilai <i>packet loss</i> protokol <i>routing</i> AOMDV; (b) grafik perbandingan nilai <i>packet loss</i> protokol <i>routing</i> OLSR; (c) grafik perbandingan nilai <i>packet loss</i> protokol <i>routing</i> ZRP;.....	35
Gambar 4.6 Perbandingan nilai rata-rata <i>throughput</i> . (a) grafik perbandingan nilai <i>throughput</i> protokol <i>routing</i> AOMDV; (b) grafik perbandingan nilai <i>throughput</i> protokol <i>routing</i> OLSR; (c) grafik perbandingan nilai <i>throughput</i> protokol <i>routing</i> ZRP;.....	36
Gambar 4.7 Perbandingan nilai rata-rata <i>end to end delay</i> . (a) grafik perbandingan nilai <i>end to end delay</i> protokol <i>routing</i> AOMDV; (b) grafik perbandingan nilai <i>end to end delay</i>	

protokol *routing* OLSR; (c) grafik perbandingan nilai *end to end delay* protokol  
*routing* ZRP;..... 37

## DAFTAR TABEL

Tabel 2.1 Nilai Tipikal Dari Eksponen <i>Path Loss</i> $\beta$ .....	14
Tabel 2.2 Nilai Khas Dari <i>Shadowing Deviation</i> Dengan dB.....	14
Tabel 3.1 Parameter Simulasi.....	21
Tabel 3.3 Jadwal Pelaksanaan Kegiatan.....	24
Tabel 4.1 Kinerja berdasarkan persentase <i>packet loss</i> .....	33
Tabel 4.2 Kinerja berdasarkan <i>throughput</i> .....	34
Tabel 4.3 Kinerja berdasarkan <i>end to end delay</i> .....	34

## ABSTRAK

VANET merupakan *subset Mobile Adhoc Network* (MANET) yang memanfaatkan teknologi *wireless*, yang dimana pada jaringan VANET *node* dipresentasikan sebagai kendaraan sedangkan rute untuk kendaraan dipresentasikan sebagai jalan raya yang akan dilalui oleh kendaraan tersebut. Pada umumnya pada VANET terdapat 3 klasifikasi protokol *routing* yaitu reaktif, proaktif, dan *hybrid*. Kinerja protokol *routing* sangat dipengaruhi oleh model propagasi [1]. Propagasi pada jaringan merupakan perilaku atau proses perambatan gelombang radio, yang dalam hal ini berfungsi sebagai media transmisi dari pengirim (*transmitter*) ke tujuan (*receiver*). Pada penelitian ini akan dilakukan analisis variasi model propagasi yakni *FreeSpace*, *TwoRayGround*, *Shadowing*, dan *Nakagami* terhadap kinerja protokol *routing* AOMDV (reaktif), OLSR (proaktif), dan ZRP (*hybrid*). Kinerja protokol *routing* tersebut akan diuji menggunakan parameter pengujian *packet loss*, *end to end delay*, dan *throughput*.

**Kata Kunci :** VANET, AOMDV, OLSR, ZRP, *FreeSpace*, *TwoRayGround*, *Shadowing*, *Nakagami*.

## **ABSTRACT**

*VANET is a subset of Mobile Adhoc Network (MANET) that utilizes wireless technology, which in the VANET network the node is presented as a vehicle and the route for a vehicle is presented as a highway that will be passed by that vehicle. In general, VANET has 3 classification routing protocols, namely reactive, proactive, and hybrid. Performance of routing protocol is greatly influenced by the propagation model. Propagation on the network is the behavior of radio waves, which in this case is used as a transmission medium from the transmitter to the destination (receiver). This research will analyze the variation of propagation models namely FreeSpace, TwoRayGround, Shadowing, and Nakagami on the performance of AOMDV (reactive), OLSR (proactive), and ZRP (hybrid). The performance of the routing protocol will be tested using test parameters including packet loss, end to end delay, and throughput.*

**Keywords:** *VANET, AOMDV, OLSR, ZRP, FreeSpace, TwoRayGround, Shadowing, Nakagam*

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Teknologi informasi saat ini mengalami perkembangan cukup cepat, seiring dengan hal tersebut telah banyak penelitian yang dilakukan untuk mencari alternatif jalan keluar untuk masalah sehari-hari. Salah satu masalah yang banyak ditemukan terutama di kota-kota khususnya di jalan raya adalah kemacetan, kecelakaan lalu lintas, pencarian rute alternatif, dan lain-lain, banyak penelitian yang dilakukan mencari solusi masalah ini, salah satunya adalah dengan memanfaatkan teknologi *Vehicle Adhoc Network* (VANET), dengan VANET kendaraan-kendaraan dapat bertukar informasi mengenai kondisi lalu lintas jalan raya.

Berdasarkan data Badan Pusat Statistik (BPS) Kota Bima tahun 2017 jumlah kendaraan baik kendaraan bermotor maupun tidak bermotor yang terdapat di Kota Bima adalah sebanyak 51.269 unit, dengan jumlah kendaraan sebanyak itu akan berpotensi menimbulkan kemacetan serta kecelakaan lalu lintas. Sehingga diperlukan suatu solusi jalan keluar untuk mengatasi permasalahan-permasalahan tersebut, oleh sebab itu lalu lintas Kota Bima dinilai cocok apabila dijadikan sebagai objek pada penelitian ini.

VANET merupakan *subset Mobile Adhoc Network* (MANET) yang memanfaatkan teknologi *wireless*, yang dimana pada jaringan VANET *node* direpresentasikan sebagai kendaraan sedangkan rute untuk kendaraan direpresentasikan sebagai jalan raya yang akan dilalui oleh kendaraan tersebut. VANET mempunyai karakteristik jaringan dengan mobilitas tinggi dan pergerakan *node* yang tidak dapat ditentukan, oleh karenanya topologi akan mengalami perubahan secara cepat. Di dalam VANET terdapat 3 klasifikasi protokol jaringan yaitu protokol *routing* reaktif, proaktif, dan *hybrid*. Kinerja VANET juga sangat dipengaruhi oleh model propagasi [1]. Propagasi pada jaringan merupakan perilaku atau proses perambatan gelombang radio, yang dalam hal ini berfungsi sebagai media transmisi dari pengirim (*transmitter*) ke tujuan (*receiver*).

Tujuan penelitian ini adalah melihat gambaran bagaimana pengaruh variasi model propagasi antara lain adalah *Free Space*, *TwoRayGround*, *Shadowing*, dan *Nakagami* terhadap protokol-protokol *routing Ad Hoc on Demand Multipath Distance Vector* (AOMDV) yang mewakili jenis protokol *routing* reaktif, *Optimized Link State Routing* (OLSR) yang mewakili jenis protokol *routing* proaktif, dan *Zone Routing Protocol* (ZRP) yang mewakili jenis protokol *routing hybrid* jika diterapkan pada peta jalan Kota Bima. Dari masing-masing ketiga protokol *routing* tersebut yakni

AOMDV, OLSR, dan ZRP akan diimplementasikan jenis-jenis variasi model propagasi dan dilihat bagaimana pengaruh variasi model propagasi tersebut terhadap kinerja masing-masing protokol *routing* apabila diterapkan pada peta jalan kota Bima.

AOMDV merupakan protokol *routing* reaktif yang memiliki unjuk kerja yang lebih bagus bila dibandingkan dengan protokol *routing* reaktif jenis lain seperti AODV apabila menggunakan MAC *type* IEEE 802.11p yang dimana MAC *type* IEEE 802.11p ini dinilai cocok apabila digunakan pada jaringan VANET [2]. OLSR merupakan protokol *routing* proaktif yang memiliki performa yang lebih baik bila dibandingkan dengan protokol *routing* jenis proaktif lain dalam jaringan VANET seperti DSDV dalam parameter uji *packet delivery ratio*, *end to end delay*, dan *routing overhead* [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].

Parameter uji kinerja yang digunakan pada penelitian ini adalah *packet loss* yaitu jumlah paket yang hilang selama proses simulasi, *end to end delay* yaitu waktu yang diperlukan selama pengiriman paket data dari sumber ke tujuan, dan *Troughput* yaitu jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. Pada penelitian ini akan dilakukan simulasi jaringan VANET menggunakan *tools software Simulator Network 2* (NS2) dan *Simulation for Urban Mobility* (SUMO). Rute yang digunakan adalah peta jalan Kota Bima yang diambil pada situs *web OpenStreetMap*. Hasil yang diharapkan dari penelitian ini adalah bagaimana pengaruh model propagasi terhadap protokol *routing* reaktif, proaktif, *hybrid* pada jaringan VANET.

## 1.2 Rumusan Masalah

Adapun rumusan masalah yang diangkat pada penelitian ini adalah “Bagaimana pengaruh beberapa variasi model propagasi terhadap kinerja protokol *routing* reaktif, proaktif, dan *hybrid* di lingkungan VANET khususnya pada peta jalan Kota Bima?”

## 1.3 Batasan Masalah

Ruang lingkup permasalahan dari penelitian ini adalah sebagai berikut:

1. Jaringan yang digunakan adalah VANET dengan peta jalan Kota Bima
2. Jenis variasi propagasi yang digunakan adalah *Free Space*, *TwoRayGround*, *Shadowing*, dan *Nakagami*
3. Protokol *routing* yang digunakan adalah AOMDV (reaktif), OLSR (proaktif), dan ZRP (*hybrid*)

4. *Software* simulasi yang digunakan adalah *Network Simulation 2 (NS-2)* versi 2.35 dan *Simulation for Urban Mobility (SUMO)*
5. Tipe koneksi yang digunakan adalah UDP
6. Tidak membahas tentang keamanan jaringan
7. Komunikasi antara *Vehicle to Vehicle (V2V)*

#### **1.4 Tujuan Penelitian**

Adapun tujuan dilakukannya penelitian ini adalah untuk mengetahui pengaruh beberapa variasi model propagasi terhadap kinerja protokol *routing* reaktif, proaktif, dan *hybrid* di lingkungan VANET khususnya pada peta jalan Kota Bima

#### **1.5 Manfaat Penelitian**

Setelah dilakukannya proses penelitian ini diharapkan dapat memberikan gambaran bagaimana dampak berbagai variasi model propagasi terhadap protokol *routing* reaktif, proaktif, dan *hybrid* pada jaringan VANET pada peta jalan Kota Bima.

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Pada penelitian [5] oleh Khalid Abdel Hafeez, Lian Zhao, Zaiyi Liao, dan Bobby Ngok-Wah Ma, 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 tadi. 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*).

Pada penelitian [6] oleh Dhiya'an Sabila Ramadhani dan Radityo Anggoro, dilakukan sebuah studi perbandingan model propagasi *TwoRayGround* dan *Nakagami* menggunakan protokol *routing* OLSR dengan parameter simulasi menggunakan 50 *node*, kecepatan maksimal *node* 5 m/s, 10 m/s, 15 m/s serta luas area simulasi adalah 510x510 meter. Pada penelitian ini dilakukan perbandingan antara model propagasi menggunakan protokol *routing* OLSR dengan parameter uji yang digunakan adalah *packet delivery ratio* (PDR), *end to end delay*, dan *routing overhead*. Hasil yang didapat pada penelitian ini adalah pada parameter uji *packet delivery ratio* (PDR) performa model propagasi *Nakagami* lebih baik dibandingkan dengan *TwoRayGround*, hal ini dikarenakan kemampuan model propagasi *Nakagami* untuk mengidentifikasi lingkungan sekitar dan pengaruh ketinggian antenna. Untuk parameter uji *end to end delay* propagasi *Nakagami* kembali unggul bila dibandingkan dengan *TwoRayGround* hal ini dikarenakan mobilitas pengiriman paket yang dinamis pada model propagasi *Nakagami*, hal ini memungkinkan terjadinya rute yang terputus saat pengiriman paket data sehingga paket pengiriman akan dimasukkan ke dalam antrian untuk menunggu terbentuknya rute baru sebelum pengiriman kembali dilakukan. Sedangkan untuk parameter uji *routing overhead* model propagasi *TwoRayGround* memiliki nilai yang lebih besar

dibandingkan dengan *Nakagami*, hal ini terjadi karena pengiriman paket *routing* jenis *send* dan *forward* yang dilakukan oleh *TwoRayGround* lebih banyak dari *Nakagami*.

Pada penelitian [7] oleh Ramesh C. Poonia dan Vikram Singh, 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 m, jumlah variasi *node* yang dipakai adalah 10, 50, 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 [8] oleh Bhargav Makodia dan Dhruvi Marsonia, dilakukan perbandingan beberapa variasi model propagasi yaitu *Free Space*, *TwoRayGround*, dan *Shadowing* pada protokol *routing* AODV, 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 [2] oleh Prof. Shivendu Dubey, Anurag Nema dan Prof. Ashok Verma, dilakukan perbandingan terhadap jenis protokol *routing* reaktif yaitu protokol *routing* AODV dan AMODV pada jaringan VANET, jumlah *node* yang digunakan adalah sebanyak 30, 50, dan 70 *node* dengan kecepatan pergerakan *node* adalah 90 km/h, menggunakan jenis propagasi *TwoRayGround* pada area simulasi sebesar 800x800 m<sup>2</sup>, serta MAC Type yang digunakan adalah 802.11 dan 802.11p. Kinerja masing-masing protokol *routing* diukur menggunakan parameter-parameter QOS seperti *average throughput*, *normalized routing load* (NRL), dan *drop packets*. Hasil dari penelitian

ini menyimpulkan bahwa protokol *routing* AOMDV berkinerja lebih baik pada standar MAC type 802.11p pada parameter *Throughput*, *NRL*, dan *Drop Packet*.

Pada penelitian [3] Made Dia Agustya, dilakukan perbandingan terhadap dua jenis protokol *routing* proaktif yakni OLSR dan DSDV pada jaringan VANET menggunakan model propagasi *TwoRayGround*, dengan jumlah variasi *node* adalah sebanyak 25, 50, 75, dan 100 *node* dengan kecepatan maksimal *node* adalah 15 m/s pada area simulasi 1500 m x 1500 m, menggunakan peta *grid* dan *real* (lingkungan perkotaan lalu lintas kota Surabaya). Kinerja masing-masing protokol *routing* diukur menggunakan parameter QOS seperti *packet delivery ratio*, *end to end delay*, dan *routing overhead*. Hasil kesimpulan pada penelitian tersebut mengatakan bahwa rata-rata protokol *routing* OLSR mempunyai kinerja yang lebih baik bila dibandingkan dengan DSDV.

Pada penelitian [4] oleh Aji Setiabudi, Amalia ayu Pratiwi, Ardiansyah, Doan.Permana, dan Riri Fitri Sari, dilakukan 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 m<sup>2</sup>. 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*, *PDR*, *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.

Apabila ditarik suatu kesimpulan secara garis besar pada penelitian-penelitian yang telah dilakukan tersebut hanya menggunakan satu jenis klasifikasi protokol *routing* saja yang pada umumnya terdapat tiga jenis protokol *routing*. Sedangkan pada penelitian ini menggunakan tiga jenis klasifikasi protokol *routing* yakni reaktif, proaktif, dan *hybrid*. Kemudian selanjutnya adalah penggunaan jenis model propagasi, yang dimana pada penelitian-penelitian sebelumnya hanya menggunakan satu sampai tiga jenis model propagasi, pada penelitian ini menggunakan empat jenis model propagasi yaitu *FreeSpace*, *TwoRayGround*, *Shadowing*, dan *Nakagami*.

## 2.2 Dasar Teori

Pada sub bab ini akan dijelaskan teori-teori dasar tentang jaringan komputer serta teori –teori tentang lingkup jaringan VANET.

### 2.2.1 Jaringan Komputer

Jaringan komputer adalah himpunan interkoneksi antara dua komputer atau lebih yang terhubung dengan media kabel (*wired*) atau tanpa kabel (*wireless*). Dua buah komputer dikatakan terkoneksi apabila keduanya bisa saling bertukar data atau informasi dan berbagai sumber daya (*resource*) yang dimiliki.

Tujuan jaringan komputer yaitu:

1. Membagi sumber daya, contohnya berbagi pemakaian *printer*, CPU, memori.
2. Komunikasi, contohnya *e-mail*, *instant messaging*, *chatting*.
3. Akses informasi, contohnya *web browsing*

Agar dapat mencapai tujuan yang sama, setiap bagian dari jaringan komputer meminta dan memberikan layanan (*service*). Pihak yang meminta layanan disebut *client* dan yang memberikan layanan disebut *server*. Arsitektur ini disebut dengan sistem *client-server*, dan digunakan hampir di setiap aplikasi jaringan komputer.

### 2.2.2 Jaringan Nirkabel *Ad-Hoc* (*Ad-Hoc Wireless Network*)

Jaringan ini merupakan jaringan tanpa kabel, dimana komunikasi data dilakukan media transmisi gelombang radio, khususnya transmisi radio standar IEEE 802.11. Jaringan ini dikatakan *ad-hoc* karena setiap *node* akan meneruskan paket data ke *node* yang lain, sehingga terbentuk sebuah jaringan dengan koneksi yang dinamis.

Sangat berbeda dengan jaringan kabel dimana sebuah *router* bertugas meneruskan paket data ke *node* yang lain. Begitu juga dengan jaringan *nirkabel* infrastruktur, dimana ada *node* khusus yang dikenal dengan *access point* akan mengatur komunikasi di antara *node-node* dalam jaringan.

Karakteristik jaringan *nirkabel ad-hoc* yang tersentralisasi, membuat jaringan ini cocok untuk beragam aplikasi, dimana tidak dibutuhkan *node* sentral dalam implementasinya. Selain itu skala dalam jaringan ini dapat ditingkatkan dibandingkan dengan jaringan infrastruktur yang kapasitasnya terbatas sesuai dengan apa yang sudah diidentifikasi sebelumnya. Konfigurasi yang minim dan pengembangan yang relatif cepat membuat jaringan *ad-hoc* cocok untuk situasi yang mendesak seperti bencana alam atau konflik

militer. Adanya kehadiran dari protokol *routing* yang dinamis dan adaptif, jaringan *ad-hoc* akan mampu dibentuk secara cepat.

### **2.2.3 Vehicular Ad-Hoc Network (VANET)**

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/perlambatan, kondisi jalan.

### **2.2.4 Protokol Routing**

*Routing* adalah suatu proses memindahkan informasi dari pengirim ke penerima melalui sebuah jaringan (Cisco, 2004). Sedangkan Protokol merupakan aturan yang mengatur setiap komputer untuk saling bertukar informasi melalui media jaringan. Sehingga *Routing Protocol* diperlukan untuk mengatur bagaimana *router* berkomunikasi antara satu dengan yang lain dalam menyebarkan informasi, yang memungkinkan *router* untuk memilih rute pada jaringan

komputer (Kopp & Hons, 2002). *Routing protocol* untuk jaringan ad-hoc dibagi menjadi tiga tipe yaitu Proaktif, Reaktif, dan Hybrid.

### **2.2.5 Protokol Routing Reaktif**

Protokol *routing* reaktif bersifat *on-demand*, artinya membentuk sebuah rute dari satu *node* sumber ke *node* tujuan hanya berdasarkan pada permintaan *node* sumber tersebut. Sehingga proses pencarian *route* hanya dilakukan apabila *node* sumber membutuhkan komunikasi dengan *node* tujuan. Pada *routing table* yang dimiliki oleh sebuah *node* berisikan informasi *route* ke *node* tujuan saja. Contoh *on demand routing* adalah AODV (*Ad Hoc On-Demand Distance Vector*), AOMDV (*Ad Hoc on Demand Multipath Distance Vector*), DSR (*Dynamic Source Routing*), TORA (*Temporally Ordered Routing Algorithm*), SSR (*Signal Stability Routing*), dan ASR (*Associativity Based Routing*).

### **2.2.6 Protokol Routing Proaktif**

Protokol *routing* proaktif bersifat tabel *driven* artinya dimana setiap *node* menyimpan tabel yang berisi informasi rute ke setiap *node* yang diketahuinya, artinya sebuah *node* mengetahui semua *route* ke *node* lain yang berada dalam jaringan tersebut. Informasi rute diperbaharui secara berkala jika terjadi perubahan *link*. Penggunaan protokol *routing* proaktif secara mendasar memberikan solusi terpendek *end to end delay*, karena informasi *routing* selalu tersedia dan diperbaharui secara berkala dibandingkan protokol *routing* reaktif. Kekurangan dari protokol *routing* proaktif adalah terlalubanyak penggunaan sumber daya seperti *overhead* saat memperbaharui informasi *routing*. Adapun contoh tabel *driven routing* adalah OLSR (*Optimized Link State Routing*), DSDV (*Destination Sequenced Distance Vector*), CGSR (*Clusterhead Gateway Switch Routing*), dan WRP (*Wireless Routing Protocol*).

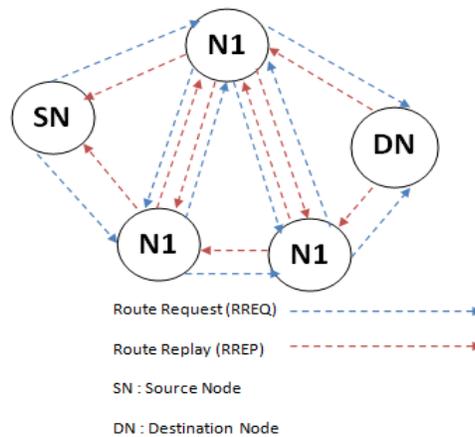
### **2.2.7 Protokol Routing Hybrid**

Protokol *Routing Hybrid* merupakan gabungan antara prinsip kerja Protokol *Routing* reaktif dan proaktif, maksudnya adalah ada suatu kondisi yang dimana menggunakan prinsip kerja protokol *routing* reaktif, dan ada pula suatu kondisi diimplementasikannya prinsip dari protokol *routing* proaktif. Contoh dari protokol *routing hybrid* adalah *Zone Routing Protocol* (ZRP).

### **2.2.8 Protokol Routing Ad Hoc on Demand Multipath Distance Vector (AOMDV)**

AOMDV adalah protokol *routing* perkembangan dari protokol AODV. Jumlah rute yang ditemukan setiap kali melakukan pencarian rute adalah perbedaan utama antara AODV dan AOMDV. AOMDV dan AODV menggunakan sebuah sistem *sequence number* untuk

memastikan bahwa rute yang dihasilkan adalah *loop-free* serta memiliki informasi *routing* yang paling terbaru. Pada AOMDV dan AODV, terdapat tiga buah pesan utama yang digunakan untuk proses pembentukan jalur *routing* dan pemeliharaan jalur *routing* yaitu : *route request* (RREQ), *route replay* (RREP) dan *route error* (RERR) (Awerbuch & Mishra,2014). Namun AOMDV pada saat pencarian rute tidak seperti AODV yang hanya memilih satu RREP, tetapi pada AOMDV setiap RREP akan dipertimbangkan oleh *node* asal sehingga beberapa path bisa ditemukan dalam satu pencarian rute. Dengan demikian, jika terjadi kegagalan rute pada saat perjalanan maka dapat dialihkan kerute yang lain.



Gambar 2.1 Ilustrasi Topologi AOMDV

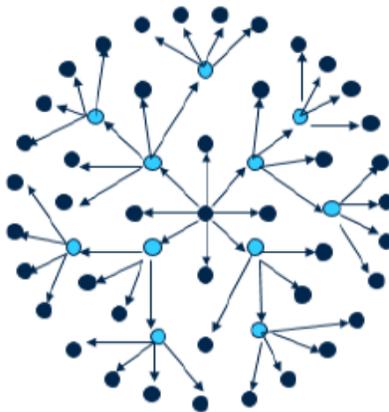
Gambar 1 memperlihatkan langkah langkah protokol AOMDV untuk melakukan pencarian rute dan pemeliharaan rute. SN akan membanjiri jaringan dengan paket RREQ ketika ingin melakukan komunikasi dengan *node* tujuan sehingga *node* lain akan mendapatkan beberapa salinan dari RREQ yang sama. Selanjutnya, semua Salinan tersebut diperiksa untuk membuat rute alternatif, namun pemeliharaan rute hanya dibuat menggunakan RREQ yang dapat mempertahankan *loop-freedom* dan *disjointness* mulai dari *node* asal. Ketika intermediate *node* menerima rute pemeliharaan melalui salinan RREQ, node ini akan memeriksa apakah ada satu atau lebih *forward paths* ke *node* tujuan yang valid. Jika ada, *node* ini akan membuat paket RREP dan mengirim kembali melalui rute pemeliharaan ke *node* sumber.

Saat *node* tujuan menerima salinan RREQ, *node* tersebut juga membuat rute pemeliharaan dengan cara yang sama dengan yang dilakukan oleh *intermediate node*. Namun, RREP yang dibuat oleh *node* tujuan dibuat dengan aturan yang lebih “longgar”. Maksudnya adalah *node* tujuan bisa mengirim RREP melalui rute pemeliharaan yang *loop-free* tanpa harus *disjoint*. Hal ini dilakukan untuk mencegah “*route cutoff*” atau rute yang dihapus karena terjadi

*suppressing* atau ketika sebuah *node* harus memilih satu dari dua atau lebih path. *Route maintenance* pada AOMDV adalah penambahan sederhana pada AODV. Sama seperti AODV, AOMDV menggunakan paket RERR. Sebuah *node* akan membuat atau meneruskan paket RERR untuk *node* tujuan saat path terakhir ke *node* tujuan rusak. AOMDV juga melakukan optimalisasi untuk menyelamatkan paket yang sedang dikomunikasikan melalui link yang rusak dengan meneruskan ulang paket tersebut melalui jalur alternatif.

### 2.2.9 Protokol *Routing Optimized Link State Routing (OLSR)*

OLSR merupakan protokol *routing* proaktif, yang dapat dengan segera menyediakan *routing* ke semua *network* tujuan yang ada. Protokol ini merupakan pengembangan dari algoritma *link-state* untuk memenuhi persyaratan dari jaringan nirkabel dinamis seperti MANET. OLSR merupakan protokol *routing* yang menggunakan algoritma *link-state* klasik dan algoritma Dijkstra untuk mencari *shortest path*. Optimalisasi ini berdasarkan pada konsep *Multipoint Relays (MPR)*. Setiap *node* menyeleksi *node-node* tetangganya sebagai *Multipoint Relay (MPR)*. Pada OLSR, hanya *node* yang berperan sebagai MPR yang bertanggung jawab untuk melanjutkan *control traffic* (paket kontrol), yang dimaksud untuk penyebaran ke seluruh jaringan. *Multipoint Relays (MPRs)* menyediakan mekanisme untuk melakukan *flood control traffic* dengan mengurangi jumlah transmisi yang dibutuhkan

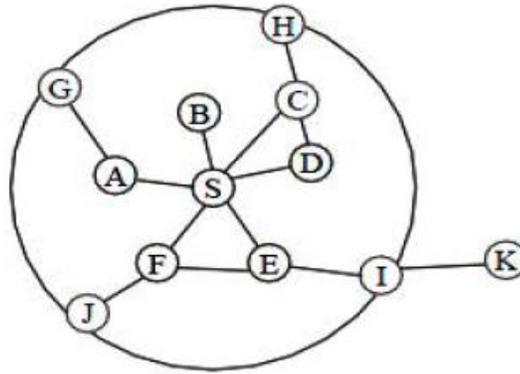


Gambar 2.2 Ilustrasi Topologi OLSR

### 2.2.10 Protokol *Routing 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.



Gambar 2.3 Ilustrasi Topologi ZRP

Pada Gambar 3, sebuah zona *routing* dengan *node* sentral S dan jumlah *hop* pada zona *node* S adalah dua. *Node* yang masuk ke dalam zona *routing node* S adalah *node* A-J, kecuali *node* K. *Node* A-F berjarak lebih kecil dari jumlah *hop*. Sedangkan *node* G-J berjarak sama dengan jumlah *hop*. *Node* H dan J dapat dicapai melalui dua jalur, yang satu berjarak dua *hop* dan yang satu lagi berjarak tiga *hop*. *Node* H dan J masih termasuk ke dalam *node* S, karena jalur terpendeknya lebih kecil atau sama dengan jumlah *hop*.

### 2.2.11 Propagasi Gelombang Radio

Propagasi adalah rambatan gelombang *microwave* melalui udara dari antenna pemancar ke antenna penerima yang jaraknya bisa mencapai ribuan kilometer. Media perambatan atau biasa juga disebut saluran transmisi gelombang dapat berupa fisik yaitu sepasang kawat konduktor, kabel koaksial dan berupa non fisik yaitu gelombang radio atau sinar laser.

Propagasi gelombang radio merupakan sesuatu yang sangat penting untuk mengetahui dan mengerti rintangan dan gangguan dalam lingkungan radio bergerak. Pengetahuan terhadap propagasi gelombang radio juga sangat penting dalam perencanaan dan pengoperasian komunikasi dengan gelombang radio agar komunikasi dapat berjalan dengan lancar.

Sistem komunikasi bergerak atau biasa dikenal dengan sistem *cellular* menggunakan gelombang radio sebagai media untuk mengkoneksikan antara perangkat satu dengan yang lain. Pada konsep komunikasi dengan gelombang radio ini dikenal suatu mode propagasi. Propagation model atau model propagasi adalah suatu cara untuk memprediksi daya sinyal rata-

rata. Ada tiga mekanisme dominan dari propagasi pada sistem komunikasi bergerak (*mobile*) yaitu pantulan (*reflection*), difraksi, dan hamburan (*scattering*).

### 2.2.12 Model Propagasi *Free Space*

Model Propagasi *Free Space* merupakan model paling pertama yang digunakan dalam MANET oleh para peneliti. Model propagasi *Free Space* mengasumsikan kondisi propagasi ideal bahwa hanya terdapat parameter *line-of-sight* (LOS) antar *transmitter* dan *receiver*. Pada persamaan 1 di menunjukkan bahwa sinyal merambat melalui ruang terbuka tanpa dipengaruhi oleh dampak dari lingkungan sekitar.

$$P(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \dots\dots\dots (1)$$

Pada persamaan di atas  $P(d)$  adalah daya sinyal yang diterima dalam Watt.  $P_t$  adalah kekuatan sinyal yang ditransmisikan.  $G_t$  dan  $G_r$  adalah gain dari antenna *transmitter* dan *receiver*.  $\lambda$  merupakan panjang gelombang,  $L$  adalah *system loss*, dan  $d$  adalah jarak antar *transmitter* dan *receiver*.

### 2.2.13 Model Propagasi *TwoRayGround*

Model propagasi *TwoRayGround* lebih realistis bila dibandingkan dengan model propagasi *Free Space* ketika hendak mempertimbangkan *reflection ground path* antara *transmitter* dan *receiver*. Model propagasi ini cukup baik untuk mengkalkulasikan daya yang diterima pada jarak yang jauh dari pemancar. Daya yang diterima pada jarak  $d$  ditunjukkan oleh

$$P_r(d) = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4 L} \dots\dots\dots (2)$$

Pada persamaan ini  $h_t$  dan  $h_r$  masing-masing adalah ketinggian antenna *transmitter* dan *receiver*.

### 2.2.14 Model Propagasi *Shadowing*

Model shadowing pada NS-2 mewujudkan model log-normal Shadowing. Diasumsikan bahwa rata-rata daya sinyal yang diterima berkurang secara logaritmik berdasarkan jarak. Variabel acak Gaussian ditambahkan ke *path loss* untuk memperhitungkan pengaruh lingkungan pada *transmitter* dan *receiver*.

Model propagasi *Shadowing* terdiri dari dua bagian, yang pertama dikenal sebagai model *path loss*, 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 sebagai berikut:

$$\frac{P_r(d_0)}{P_r(d)} = \left(\frac{d}{d_0}\right)^\beta \dots\dots\dots (3)$$

Yang dimana  $\beta$  disebut sebagai eksponen *path loss*, dan biasanya nilai tersebut secara empiris ditentukan oleh pengukuran di lapangan (area simulasi). Dalam tabel 1 diberikan beberapa nilai khas dari parameter  $\beta$  berdasarkan tipikal lingkungan

Tabel 2.1 Nilai tipikal dari eksponen *path loss*  $\beta$

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

*Path loss* dalam dB dapat dihitung menggunakan persamaan berikut

$$\left[\frac{P_r(d)}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) \dots\dots\dots (4)$$

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

$$\left[\frac{P_r(d)}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) + X_{dB} \dots\dots\dots (5)$$

Dimana  $X_{dB}$  adalah variable acak Gaussian dengan mean nol dan standar deviasi  $\sigma_{dB}$ .  $\sigma_{dB}$  dikenal sebagai *Shadowing deviation*. Pada table 2 menunjukkan beberapa nilai khas dari  $\sigma_{dB}$

Tabel 2.2 Nilai khas dari *Shadowing deviation* dengan dB

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

**2.2.15 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 sebagai berikut:

$$P_r(r) = \frac{2m^m x^{2m-1}}{\Gamma(m)\Omega^m} \exp\left[-\frac{mx^2}{\Omega}\right], x \geq 0, \Omega > 0, m \geq \frac{1}{2} \dots\dots\dots (6)$$

Dimana:

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

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

Pengaruh parameter-*m* pada model *Nakagami*:

- i. Apabila  $m \leq 1$ , maka persamaan menjadi *probability destiny function* (PDF)
- ii. Apabila  $m > 1$ , merujuk ke kanal *fading Ricean*, dimana kanal *fading Ricean* dipengaruhi oleh faktor *K*. Sehingga nilai *m* adalah

$$m = \frac{(1+k)^2}{1+2k}, k \geq 0 \dots\dots\dots (7)$$

**2.2.16 Parameter Quality of Service (QoS)**

QoS merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu *service*. QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu *service*.

QoS mengacu pada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda-beda. QoS menawarkan kemampuan untuk mendefinisikan atribut-atribut layanan jaringan yang disediakan, baik secara kualitatif maupun kuantitatif.

### 2.2.17 Throughput

*Throughput* yaitu kecepatan (*rate*) transfer data efektif, yang diukur dalam bps (*bit per second*). *Throughput* adalah jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. *Throughput* dapat didefinisikan sebagai berikut

$$\text{Throughput} = \frac{\text{Paket yang diterima}}{\text{Lama pengamatan}} \dots\dots\dots (8)$$

### 2.2.18 Packet Loss

*Packet loss* merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah *ratio* paket yang hilang dapat terjadi karena *collision* dan *congestion* pada jaringan. *packet loss* dapat didefinisikan sebagai berikut:

$$PL = \frac{(\text{Paket terkirim} - \text{Paket diterima})}{\text{Paket terkirim}} \times 100\% \dots\dots\dots (9)$$

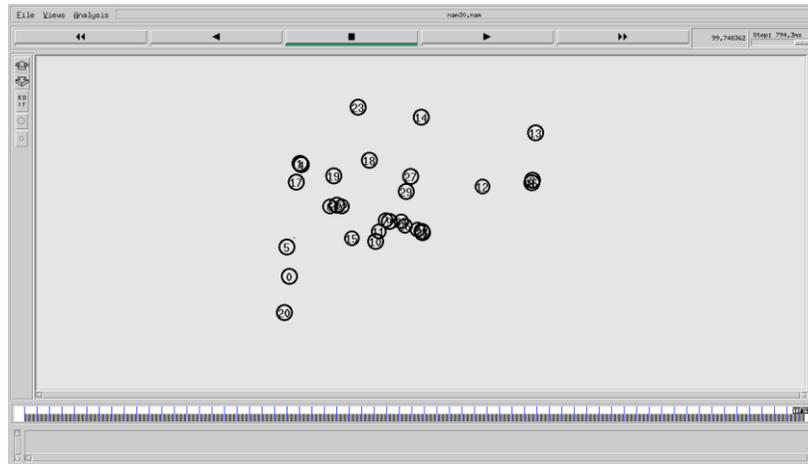
### 2.2.19 End to End Delay (E2D)

E2D dihitung dari rata-rata *delay* antara waktu paket diterima dan waktu paket dikirim. *Delay* dapat didefinisikan sebagai berikut:

$$E2D = \frac{\sum \text{waktu terima} - \text{waktu kirim}}{\text{jumlah paket terkirim}} \dots\dots\dots (11)$$

### 2.2.20 Network Simulator 2 (NS2)

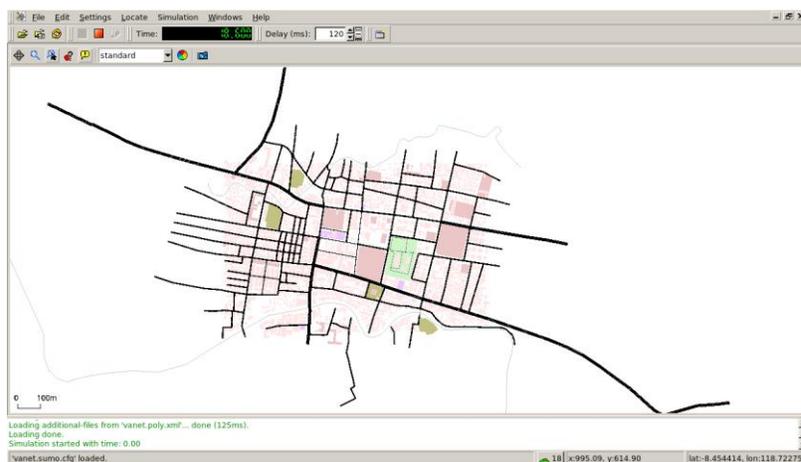
NS-2 merupakan salah satu perangkat lunak yang dapat menampilkan secara simulasi proses komunikasi dan bagaimana proses komunikasi tersebut berlangsung. NS-2 melayani simulasi untuk komunikasi dengan kabel dan nirkabel. NS-2 memiliki beberapa fitur kelebihan yang dapat dimanfaatkan dalam pemodelan dan pengujian VANET. NS-2 memiliki *tools* validasi yang berfungsi untuk menguji validitas pemodelan yang ada pada NS-2. NS-2 bersifat *open source* dibawah GPL (GNU Public Licence), sehingga dapat diunduh melalui situs NS-2 di <http://www.isi.edu/nsnam/dist>.



Gambar 2.4 Tampilan GUI NS2

### 2.2.21 Simulation of Urban Mobility (SUMO)

*Simulation of Urban Mobility* atau biasa dikenal dengan SUMO merupakan sebuah program *open source simulator* lalu lintas jalan yang memungkinkan pengguna untuk membangun simulasi pergerakan kendaraan pada topologi jaringan VANET yang disesuaikan. Pengimplementasian SUMO dimulai pada tahun 2001, dengan sebuah rilis *open source* tahun 2002, dikembangkan oleh Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker.



Gambar 2.5 Tampilan GUI SUMO

### 2.2.22 Java OpenStreetMap (JOSM)

Java *OpenStreetMap* atau biasa disingkat JOSM merupakan sebuah aplikasi *desktop* berbasis Java dan dapat dioperasikan pada sistem operasi seperti *Windows*, *Mac OS*, dan *Linux*. JOSM adalah alat penyunting bagi data *OpenStreetMap*. JOSM pertama kali dikembangkan

oleh Immanuel Scholz pada tahun 2005. Aplikasi ini tidak membutuhkan koneksi internet kala menyunting data OSM.

### **2.2.23 AWK Script**

AWK adalah sebuah bahasa pemrograman yang biasanya digunakan sebagai ekstraksi data dan alat pelaporan. AWK diciptakan di Bell Labs pada Tahun 1970, dan merupakan singkatan dari nama belakang penciptanya, yaitu Alfred Aho, Peter Weinberger, dan Brian Kernighan. AWK atau juga disebut Gawk (GNU awk). AWK biasanya digunakan untuk analisi log yang panjang atau grab teks lalu dimodifikasi dengan hanya beberapa baris kode.

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Identifikasi Kebutuhan Sistem**

Berikut merupakan identifikasi kebutuhan perangkat keras dan perangkat lunak yang akan digunakan dalam simulasi penelitian.

##### **3.1.1 Kebutuhan Perangkat Keras**

Pada penelitian ini dibutuhkan perangkat keras berupa sebuah laptop sebagai alat untuk melakukan aktifitas penelitian seperti melakukan simulasi dan pembuatan laporan dengan spesifikasi sebagai berikut:

1. CPU : Intel(R) Pentium(R) CPU P6100 @2.00GHz
2. Memori : 2 GB
3. Hardisk : HDD 320 GB

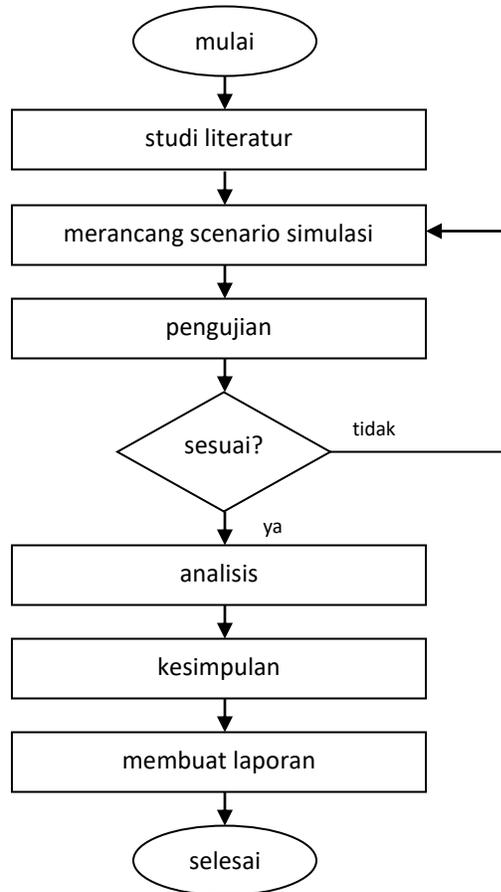
##### **3.1.2 Kebutuhan Perangkat Lunak**

Pada penelitian ini juga dibutuhkan beberapa aplikasi perangkat lunak sebagai penunjang kebutuhan selama proses penelitian berlangsung

1. Sistem operasi Linux Ubuntu 16.04 LTS 32 *bit*
2. *Network Simulator 2* (NS2) versi 2.35 yang digunakan untuk simulasi proses komunikasi antar *node* atau kendaraan
3. *Simulation of Urban Mobility* (SUMO) versi 1.0 yang digunakan untuk membangun simulasi pergerakan *node*
4. *Microsoft Excel* untuk membuat grafik hasil simulasi
5. *Java OpenStreetMap* (JSOM) untuk menyunting peta sesuai kebutuhan

#### **3.2 Diagram Alir Penelitian**

Pada diagram alir berikut ini digambarkan bagaimana proses penelitian berlangsung dari awal dimulainya proses penelitian hingga akhir. Berikut diagram alir penelitian seperti terdapat pada Gambar 3.1



Gambar 3.1 Diagram Alir Penelitian

Pada penelitian ini pertama akan dilakukan studi literatur yakni bagaimana mempelajari dan mengumpulkan fakta-fakta terkait baik melalui jurnal-jurnal penelitian maupun buku. Kemudian tahap selanjutnya adalah melakukan perancangan skenario simulasi yaitu bagaimana menentukan parameter simulasi, pembuatan peta dengan aplikasi SUMO dan rancangan alir simulasi dengan NS2 sampai menghasilkan *output* berupa *trace file* (\*.tr), selanjutnya *trace file* akan diuji pada tahap pengujian menggunakan parameter uji PDR, *Throughput*, E2ED, dan *packet loss*. Setelah tahap pengujian hasilnya akan diperiksa apakah sesuai dengan kebutuhan atau tidak, apabila tidak maka akan kembali ke proses perancangan skenario, dan apabila sesuai maka akan ke tahap selanjutnya, yaitu tahap analisis. Setelah dilakukan tahap analisis kemudian akan ditarik suatu kesimpulan serta pembuatan laporan.

### 3.3 Perancangan Skenario Simulasi

Pada penelitian ini proses perancangan skenario simulasi akan dilakukan dengan 3 tahap, yaitu tahap menentukan parameter simulasi, perancangan peta simulasi, dan perancangan alir simulasi menggunakan NS2.

### 3.3.1 Parameter Simulasi

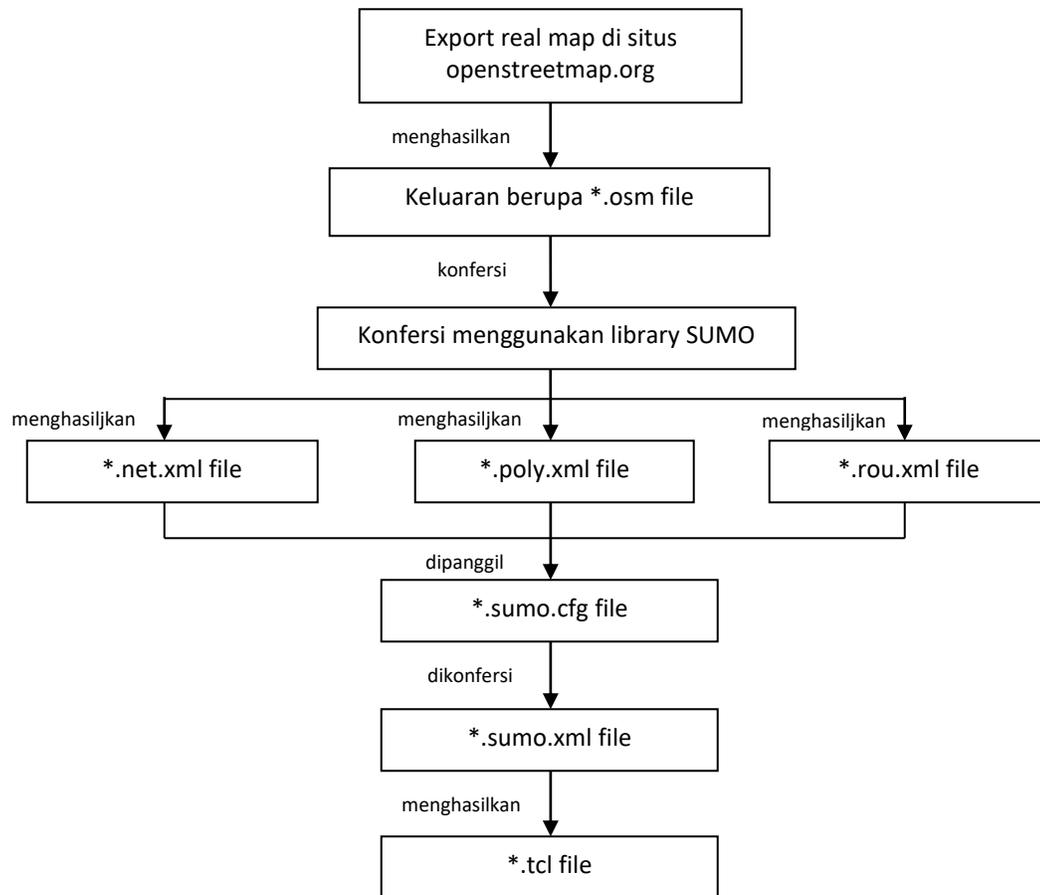
Mengacu pada penelitian [2] bahwa penelitian ini akan dilakukan dengan dengan variasi jumlah *node* sebanyak 30, 50, dan 70 dengan kecepatan *node* 90 km/h pada area simulasi sebesar 800x800 m<sup>2</sup> dengan waktu simulasi selama 100 detik. Adapun parameter simulasi selengkapnya seperti pada Tabel 3.1

Tabel 3.1 Parameter Simulasi

<b>Parameter</b>	<b>Nilai</b>
Protokol <i>Routing</i>	AOMDV, OLSR, dan ZRP
Model Propagasi	<i>Free Space, TwoRayGround, Shadowing, dan Nakagami</i>
Area Simulasi	800x800 m <sup>2</sup>
Protokol Pengiriman	UDP
Model Trafik	CBR
Jumlah <i>Node</i>	30, 50, dan 70
Kecepatan <i>Node</i>	90 km/h
Pergerakan <i>Node</i>	<i>Random Waypoint</i>
Kanal	<i>Wireless</i>
Antena	<i>OmniAntenna</i>
MAC <i>Layer</i>	IEEE 802.11p

### 3.3.2 Perancangan Peta Simulasi

Perancangan peta simulasi merupakan proses pembuatan peta simulasi studi kasus peta jalan kota Bima dari mulai melakukan pengambilan peta sampai pada proses konfersi peta tersebut menjadi \*.tcl *file* menggunakan *library* SUMO, yang dimana *file* \*.tcl tersebut akan menjadi *inputan* ketika melakukan peoses simulasi menggunakan NS-2. Berikut diagram proses perancangan peta simulasi terdapat pada gambar 3.2

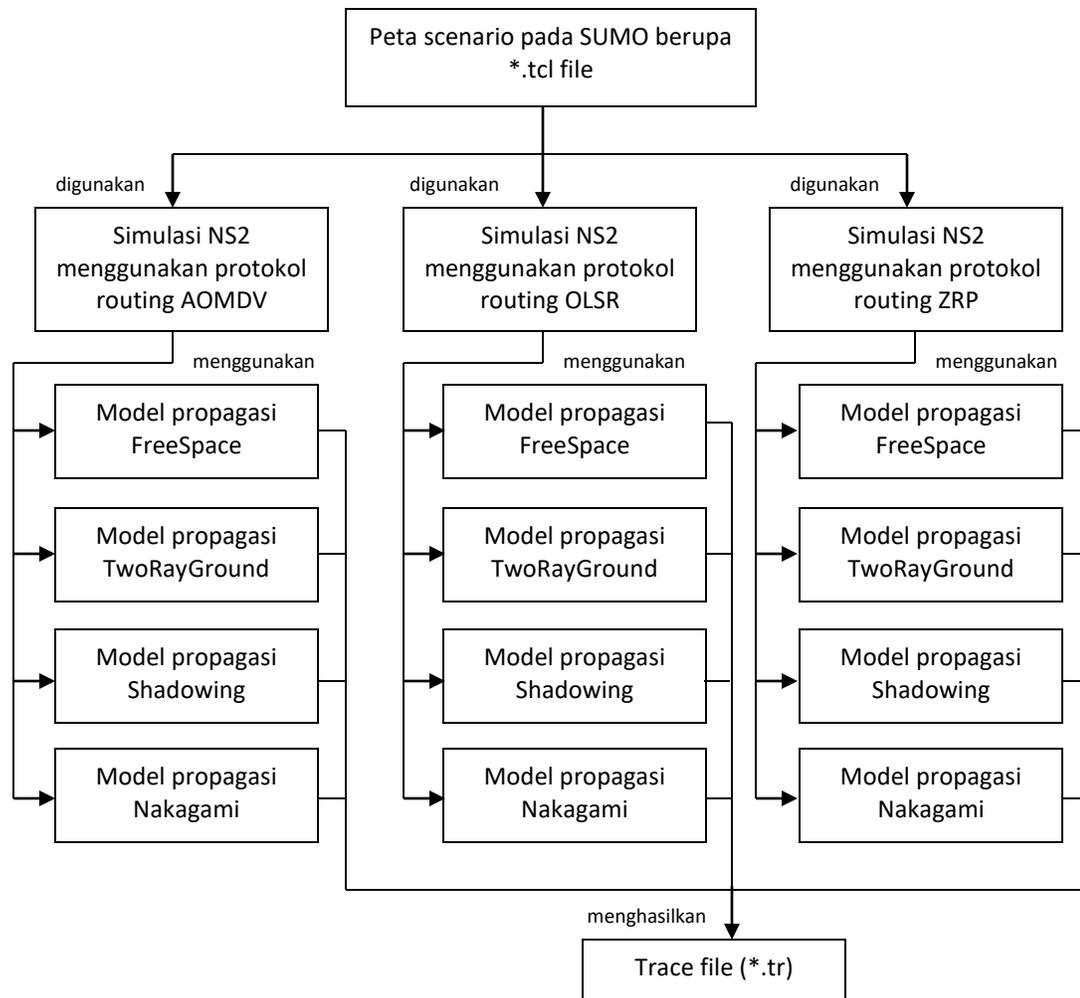


Gambar 3.2 Diagram Alir Pembuatan Peta Dengan SUMO

Perancangan peta simulasi diawali dengan mengambil area *real map* (studi kasus Kota Bima) di situs *openstreetmap.org* yang dimana akan menjadi sebuah *file* dengan ekstensi *\*.osm* (*osm file*). Selanjutnya *osm file* dikonfersi menggunakan *library SUMO* sehingga menghasilkan *file* baru yaitu *\*.net.xml*, *\*.poly.xml*, dan *\*.rou.xml* yang dimana *file-file* tersebut berisi tentang informasi jumlah *node* yang bisa ditentukan pada saat melakukan konfersi, 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 dikonfersi ke bentuk *\*.sumo.xml* dan dikonfersi lagi menjadi *\*.tcl file*

### 3.3.3 Perancangan Simulasi Dengan NS2

Proses perancangan simulasi merupakan proses perancangan protokol *routing*, model propagasi, serta memasukan parameter-parameter yang telah ditentukan ke dalam NS-2. Berikut adalah diagram alur proses simulasi pada penelitian ini seperti yang terdapat pada Gambar 3.3.



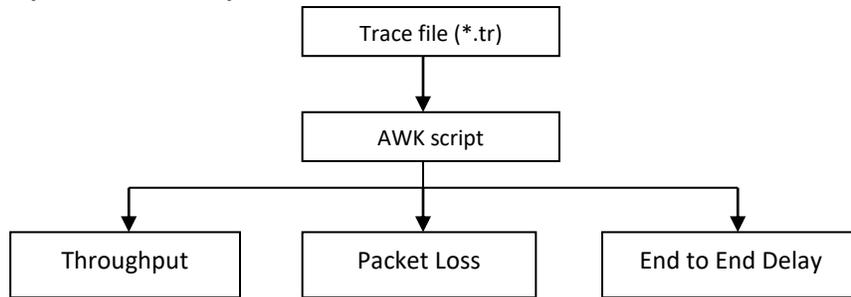
Gambar 3.3 Diagram Alir Simulasi

Perancangan simulasi akan dilakukan dengan menggunakan *software* simulasi NS2. Pada tahap ini proses simulasi akan menggunakan *\*.tcl file* yang sebelumnya telah dibuat pada proses pembuatan peta, *\*.tcl file* tersebut kemudian akan dibuat sesuai dengan kebutuhan yaitu menggunakan tiga macam variasi protokol *routing* yakni protokol *routing* AOMDV, OLSR, 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*. Dari berbagai macam variasi protokol *routing* dan model propagasi tersebut masing-masing akan menghasilkan keluaran berupa *trace file (\*.tr)*.

### 3.4 Pengujian

Tahap pengujian adalah suatu tahap dimana akan dilakukan analisis terhadap *trace file* yaitu *file* keluaran setelah dilakukannya tahap simulasi yang berisi tentang data-data hasil proses simulasi. Data-data tersebut dianalisis (*difilter*) menggunakan bahasa pemrograman AWK yang kemudian

akan menghasilkan informasi berupa *throughput*, *packet loss*, dan *end to end delay*. Setelah hasil analisis keluar maka tahap selanjutnya adalah proses perhitungan dan pembuatan grafik menggunakan *software Microsoft excel*



Gambar 3.4 Tahap Pengujian

### 3.5 Jadwal Pelaksanaan Kegiatan

Jadwal penelitian dibuat agar penelitian yang dilakukan dapat dikerjakan secara terjadwal sehingga penelitian dapat selai tepat waktu. Untuk jadwal pada penelitian ini dapat dilihat pada Tabel 3.3.

Tabel 3.3 Jadwal pelaksanaan penelitian

No	Kegiatan	Bulan 1				Bulan 2				Bulan 3				Bulan 4				Bulan 5				
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
1.	Tahap Persiapan																					
	a. Pengajuan judul	■																				
	b. Studi literatur		■	■																		
2.	Tahap pelaksanaan																					
	a. Penyusunan proposal			■	■	■	■															
	b. Asistensi laporan			■	■	■	■	■	■	■	■	■										
	c. Seminar											■										
	d. Revisi												■	■	■							
	e. Penyusunan bab IV & V													■	■							
	f. Asistensi bab IV & V													■	■	■						
	g. Ujian																				■	
	h. Revisi																				■	■

## BAB IV

### HASIL DAN PEMBAHASAN

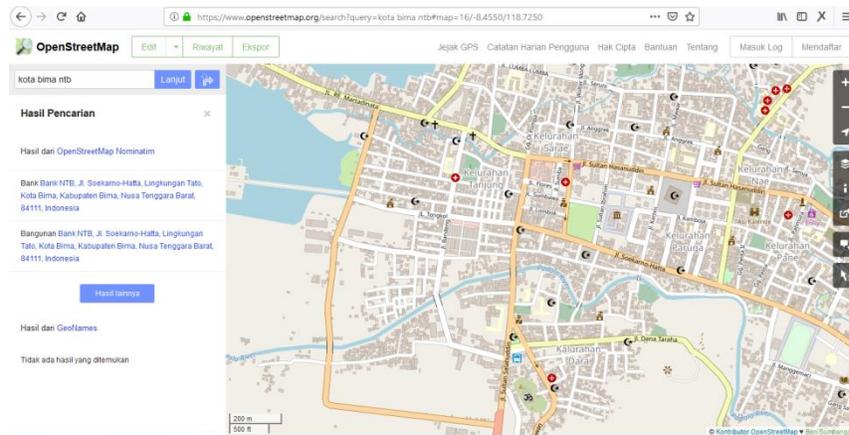
Pada bab ini akan dilakukan pembahasan terhadap hasil dari proses penelitian yang telah dilakukan, yakni meliputi pembahasan dan analisis mengenai dampak variasi model propagasi terhadap kinerja protokol *routing*.

#### 4.1 Merancang Skenario Simulasi

Pada sub bab ini akan dilakukan beberapa tahapan dalam perancangan simulasi, yaitu mulai dari tahap membuat peta simulasi, membuat rute berdasarkan peta, sampai pada pembuatan *tcl file*.

##### 4.1.1 Membuat Peta Simulasi

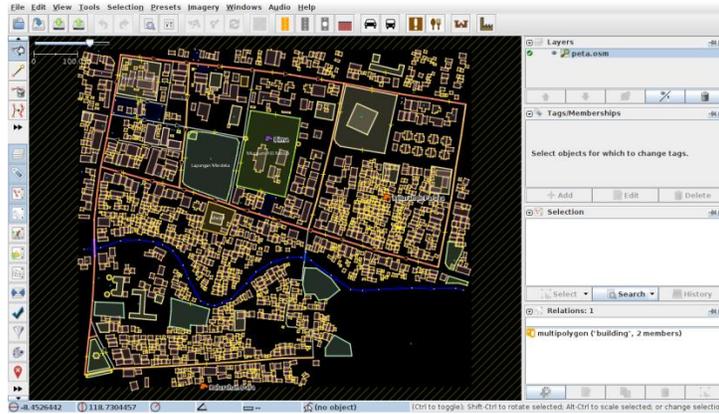
Pembuatan peta simulasi dilakukan dengan melakukan ekspor peta pada situs [openstreetmap.org](https://www.openstreetmap.org) seperti pada Gambar 4.1



Gambar 4.1 Situs Openstreetmap.org

Gambar 4.1 merupakan tampilan halaman utama dari situs [openstreetmap.org](https://www.openstreetmap.org) yang menampilkan peta kota Bima. Setelah melakukan ekspor akan menghasilkan sebuah *file* dengan ekstensi *osm* (*osm file*).

Selanjutnya adalah melakukan penyuntingan terhadap *osm file*, yaitu melakukan proses merapikan dan menghapus objek-objek yang tidak diperlukan dalam simulasi seperti gang kecil yang tidak dilewati oleh kendaraan. Proses ini menggunakan *tools* pada sebuah *software* yang bernama *Java Open Street Map* (JOSM). Setelah proses ini selesai maka tampilan peta akan tampak seperti pada Gambar 4.2



Gambar 4.2 Peta disunting menggunakan *software* JOSM

Gambar di atas merupakan tampilan peta pada *software* JOSM. Setelah proses penyuntingan selesai kemudian disimpan dengan ekstensi *.osm*.

#### 4.1.2 Membuat Rute Berdasarkan Peta (*osm file*)

Pada proses ini akan dilakukan perancangan rute berdasarkan peta jalan pada *osm file*, yaitu peta jalan kota Bima. Langkah pertama yang dilakukan adalah dengan membuka terminal pada ubuntu kemudian memasukan perintah untuk mengubah *file osm* dalam bentuk *xml file* dengan cara sebagai berikut

```
netconvert --osm-files [file osm] --lefthand -o [output/net file]
```

Selanjutnya adalah melakukan *generate* terhadap *file osm* dan *net file* menggunakan suatu *file* yang bernama *osmPolyconvert.typ.xml* yang terdapat pada direktori SUMO, proses ini bertujuan agar objek objek seperti gedung (*building*), sungai, lapangan, dan sebagainya dapat terbaca. Proses tersebut dapat dilakukan dengan perintah

```
// osmPolyconvert.typ.xml terdapat di direktori sumo-0.32.0/data/typemap/  
polyconvert --osm-files [osm file] --net-file [net file] --type-file  
osmPolyconvert.typ.xml -o [output/xml file]
```

Selanjutnya adalah membuat *file route* dengan menggunakan suatu *library* SUMO yang bernama *randomTrips.py*, proses ini dapat dilakukan dengan perintah

```
python /usr/local/src/sumo-0.32.0/tools/randomTrips.py -n [net file] -r  
[output] -e [jumlah node] -l -s [seed]
```

Selanjutnya adalah membuat sebuah *file* konfigurasi, yang dimana pada *file* inilah kemudian *file* yang telah dibuat sebelumnya dipanggil. Berikut merupakan *file* konfigurasi tersebut

```

<configuration>
  <input>
    <net-file value="map.net.xml"/>
    <route-files value="map-30-seed1.rou.xml"/>
    <additional-files value="map.poly.xml"/>
  </input>
  <time>
    <begin value="0"/>
    <end value="200"/>
    <step-length value="0.1"/>
  </time>
</configuration>

```

Langkah selanjutnya adalah mengubah *file configuration* tadi ke bentuk *file xml*. Proses ini dapat dilakukan dengan perintah.

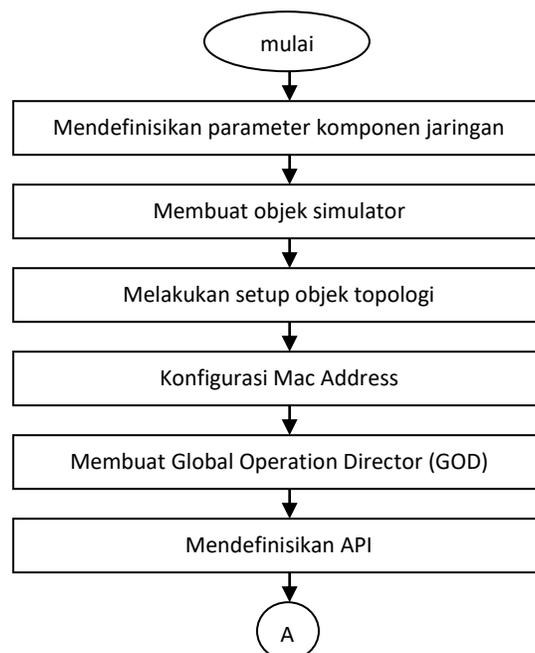
```
sumo -c [config file] --fcd-output [output/xml file]
```

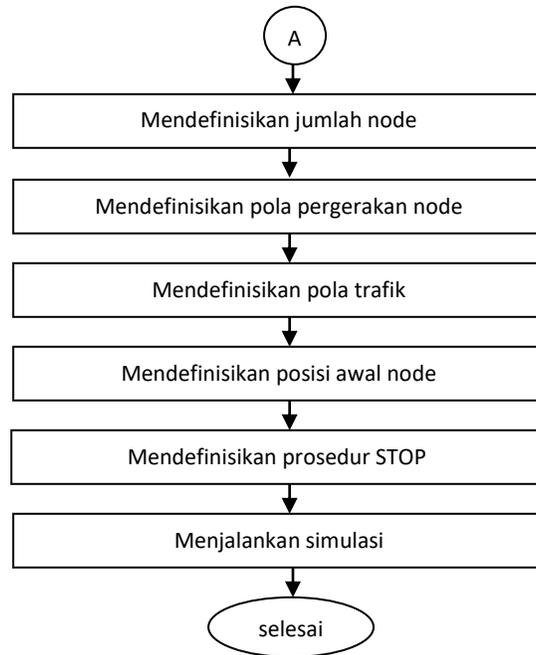
Kemudian langkah terakhir dalam membuat *file route* adalah dengan menggunakan suatu *tool* pada *library* SUMO yang bernama `traceExporter.py`, yang dimana proses ini akan menghasilkan beberapa *file tcl* yaitu *mobility.tcl* yang merupakan rute pergerakan kendaraan atau *node*, *file* yang lain adalah *map.tcl* dan *activity.tcl*. Proses ini dapat dilakukan dengan perintah

```
python /usr/local/src/sumo-0.32.0/tools/traceExporter.py --fcd-input [input/xml file] --ns2config-output [output/map.tcl] --ns2mobility-output [output/mobility.tcl] --ns2activity-output [output/activity.tcl]
```

#### 4.1.3 Pembuatan tcl File

Pada sub bab ini akan dibahas bagaimana proses pembuatan *file tcl*, yang dimana pada *file* ini akan memuat semua skenario berdasarkan parameter yang telah ditentukan. berikut merupakan diagram alir pembuatan *tcl file* seperti pada Gambar 4.3.





Gambar 4.3 Diagram alir pembuatan *file tcl*

a. Mendefinisikan parameter komponen jaringan

Tahap ini akan didefinisikan beberapa parameter komponen jaringan yang telah ditentukan untuk simulasi, komponen-komponen parameter tersebut antara lain adalah tipe kanal dan dalam penelitian ini tipe kanal yang digunakan adalah *wireless*, kemudian tipe propagasi yang digunakan yaitu *FreeSpace*, *TwoRayGround*, *Shadowing*, dan *Nakagami* yang dimana dalam contoh *script* berikut menggunakan model propagasi *Shadowing*, kemudian selanjutnya adalah penggunaan *network interface* atau kartu jaringan yaitu menggunakan tipe *wireless*, berikutnya adalah MAC layer yang digunakan pada penelitian ini adalah *MAC layer type 802.11p*, selanjutnya adalah model antena yaitu menggunakan model *OmniAntenna*, kemudian penggunaan variasi jumlah *node* yaitu 30, 50, dan 70 dan dalam contoh *script* berikut menggunakan 30 *node*, kemudian jenis protokol *routing* yaitu menggunakan AOMDV, OLSR, dan ZRP, dan dalam contoh *script* berikut menggunakan protokol *routing* OLSR, kemudian mengatur besar area simulasi dalam bidang datar yaitu sebesar 800x800 m<sup>2</sup>, dan terakhir adalah menentukan waktu simulasi yang dimana pada penelitian ini waktu simulasinya adalah 200 detik. Berikut merupakan contoh *script* untuk mendefinisikan parameter-parameter tersebut

```

set val(chan) Channel/WirelessChannel ;#Channel Type
set val(prop) Propagation/Shadowing ;# 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) 30 ;# max packet in ifq
set val(nn) 30 ;# number of mobilenodes
set val(rp) OLSR ;# routing protocol
set val(x) 800 ;
set val(y) 800 ;
  
```

```

set val(sc)           "mobility-30.tcl";
set val(cbr)         "cbr-30-seed1-rate4.tcl";
set val(stop)        200.0;

```

#### b. Membuat objek simulator

Tahap ini adalah melakukan inisialisasi terhadap *file* output yaitu berupa *trace file* (.tr) dan *file* nam (.nam). *file trace* berisi tentang informasi-informasi setelah dilakukan proses simulasi, sedangkan nam adalah *file* yang digunakan untuk menampilkan animasi simulasi pada NS2. Berikut merupakan contoh *script* untuk menginisialisasi *global variable*

```

set ns_ [new Simulator]
set tracefd [open trace-30.tr w]
$ns_ trace-all $tracefd
set namtrace [open nam-30.nam w]
$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

```

#### c. Melakukan *setup* objek topologi

Tahap ini mendefinisikan topologi berupa luas area simulasi yang akan berbentuk persegi datar (*flat grid*) berdasarkan sumbu x dan y. berikut merupakan contoh *scriptnya*

```

set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)

```

#### d. Konfigurasi *MAC Address*

Tahap ini adalah melakukan konfigurasi *MAC Address*, tipe *MAC Address* yang digunakan adalah IEEE 802.11p. *MAC Address* merupakan sebuah alamat fisik unik untuk mengidentifikasi sebuah *device* atau *node* dalam sebuah jaringan. berikut merupakan konfigurasi parameter *default* IEEE 802.11p

```

Phy/WirelessPhyExt set CStresh_ 3.162e-12 ;#-85 dBm Wireless
interface sensitivity (sensitivity defined in the standard)
Phy/WirelessPhyExt set Pt_ 0.001
Phy/WirelessPhyExt set freq_ 5.9e+9
Phy/WirelessPhyExt set noise_floor_ 1.26e-13 ;#-99 dBm for 10MHz
bandwidth
Phy/WirelessPhyExt set L_ 1.0 ;#default radio
circuit gain/loss
Phy/WirelessPhyExt set PowerMonitorThresh_ 6.310e-14 ;#-102dBm power
monitor sensitivity
Phy/WirelessPhyExt set HeaderDuration_ 0.000040 ;#40 us
Phy/WirelessPhyExt set BasicModulationScheme_ 0
Phy/WirelessPhyExt set PreambleCaptureSwitch_ 1
Phy/WirelessPhyExt set DataCaptureSwitch_ 0
Phy/WirelessPhyExt set SINR_PreambleCapture_ 2.5118; ;# 4 dB
Phy/WirelessPhyExt set SINR_DataCapture_ 100.0; ;# 10 dB
Phy/WirelessPhyExt set trace_dist_ 1e6 ;# PHY trace until
distance of 1 Mio. km ("infinty")
Phy/WirelessPhyExt set PHY_DBG_ 0
Mac/802_11Ext set CWMin_ 15
Mac/802_11Ext set CWMax_ 1023
Mac/802_11Ext set SlotTime_ 0.000013
Mac/802_11Ext set SIFS_ 0.000032
Mac/802_11Ext set ShortRetryLimit_ 7
Mac/802_11Ext set LongRetryLimit_ 4
Mac/802_11Ext set HeaderDuration_ 0.000040

```

```

Mac/802_11Ext set SymbolDuration_          0.000008
Mac/802_11Ext set BasicModulationScheme_   0
Mac/802_11Ext set use_802_11a_flag_       true
Mac/802_11Ext set RTSThreshold_           2346
Mac/802_11Ext set MAC_DBG                  0

```

*MAC Address* tipe 802.11p merupakan hasil penambahan dari tipe 802.11 yang telah disetujui oleh standar IEEE yang dimana terdapat penambahan akses nirkabel pada lingkungan *vehicular* atau kendaraan. Agar *file* tcl terlihat lebih rapih maka *file* konfigurasi tersebut dibuat terpisah dengan tcl *file*, dan cukup dipanggil dengan perintah berikut

```
source IEEE802-11p.tcl
```

#### e. Membuat *Global Operation Director* (GOD)

GOD merupakan objek yang digunakan untuk menyimpan informasi global tentang kondisi lingkungan jaringan atau *node*. Mendefinisikan variabel global bertujuan untuk mendefinisikan nilai parameter yang digunakan, sekaligus untuk memodelkan kondisi karakteristik jaringan. Contoh *scriptnya* adalah sebagai berikut

```
# Create God
create-god $val(nn)
```

#### f. Mendefinisikan API

Konfigurasi API ini terdiri dari parameter-parameter simulasi yang telah didefinisikan sebelumnya, seperti tipe protokol *routing*, propagasi, dan sebagainya. Konfigurasi API ini dapat dilakukan seperti pada contoh *script* berikut

```

$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) \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace OFF \
    -movementTrace ON \
    -channelType $val(chan)

```

#### g. Mendefinisikan jumlah *node*

Selanjutnya adalah melakukan melakukan pendefinisian *node* ke 0 sampai *node* ke n berdasarkan jumlah *node* yang telah ditetapkan pada parameter simulasi. Contoh *scriptnya* seperti berikut

```

for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]
}

```

#### h. Mendefinisikan pola pergerakan *node*

Untuk tahap ini perlu dilakukan pendefinisian terhadap variabel yang digunakan untuk memanggil *file mobility.tcl* pada bagian definisi parameter simulasi, *file mobility.tcl* adalah *file* hasil konversi dari *map osm* pada tahap sebelumnya. Berikut *script* untuk mendefinisikan tahap ini

```
puts "loading movement file..."
source $val(sc)
```

Dan berikut merupakan potongan kode isi dari *file mobility.tcl* yang mendefinisikan pola pergerakan dan posisi *node*

```
$node_(0) set X_ 855.54
$node_(0) set Y_ 603.49
$node_(0) set Z_ 0
$ns_ at 0.0 "$node_(0) setdest 855.54 603.49 0.00"
$ns_ at 0.1 "$node_(0) setdest 855.52 603.5 0.25"
```

#### i. Mendefinisikan pola trafik

Untuk melakukan pendefinisian pola trafik jaringan secara acak, dapat dilakukan dengan cara men-*generate* sebuah *file* bernama "cbrgen.tcl" pada library NS2, *file* ini terdapat pada direktori "ns-allinone-2.35/ns-2.35/indep-utils/cmu-scen-gen". berikut adalah perintah untuk men-*generate* pada proses tersebut

```
$ ns cbrgen.tcl -type [cbr] -nn [node] -seed [seed] -mc [connections] -rate [rate] > [nama file untuk menyimpan data hasil generate]
```

Dan berikut merupakan potongan isi dari *file* hasil *generate* tersebut

```
set udp_(0) [new Agent/UDP]
$ns_ attach-agent $node_(1) $udp_(0)
set null_(0) [new Agent/Null]
$ns_ attach-agent $node_(2) $null_(0)
set cbr_(0) [new Application/Traffic/CBR]
$cbr_(0) set packetSize_ 512
$cbr_(0) set interval_ 1.0
$cbr_(0) set random_ 1
$cbr_(0) set maxpkts_ 10000
$cbr_(0) attach-agent $udp_(0)
$ns_ connect $udp_(0) $null_(0)
$ns_ at 2.5568388786897245 "$cbr_(0) start"
```

#### j. Mendefinisikan posisi awal *node*

Tahap ini adalah mendefinisikan posisi awal *node* agar semua *node* dapat langsung terlihat pada saat menjalankan animasi proses simulasi. Contoh *scriptnya* sebagai berikut

```
for {set i 0} {$i < $val(nn)} {incr i} {
    $node_($i) random-motion 0
    $ns_ initial_node_pos $node_($i) 10
}
```

#### k. Mendefinisikan prosedur *stop*

Tahap ini merupakan tahap dimana didefinisikan kapan proses simulasi akan berhenti. *Scriptnya* sebagai berikut

```
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}
$ns_ at $val(stop) "stop"
```

```

$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd
    $ns_ flush-trace
    close $tracefd
}

```

### 1. Menjalankan simulasi

Tahap ini merupakan tahap terakhir dari proses pembuatan tcl *file*. *Script* untuk tahap ini sebagai berikut

```

puts "Starting Simulation..."
$ns_ run

```

## 4.2 Menjalankan Simulasi

Setelah melakukan proses pembuatan skenario simulasi, maka tahap selanjutnya adalah menjalankan simulasi *file* tcl yang telah dibuat dengan NS2, perintah yang dilakukan untuk menjalankan proses simulasi adalah seperti berikut

```

$ ns map-30.tcl

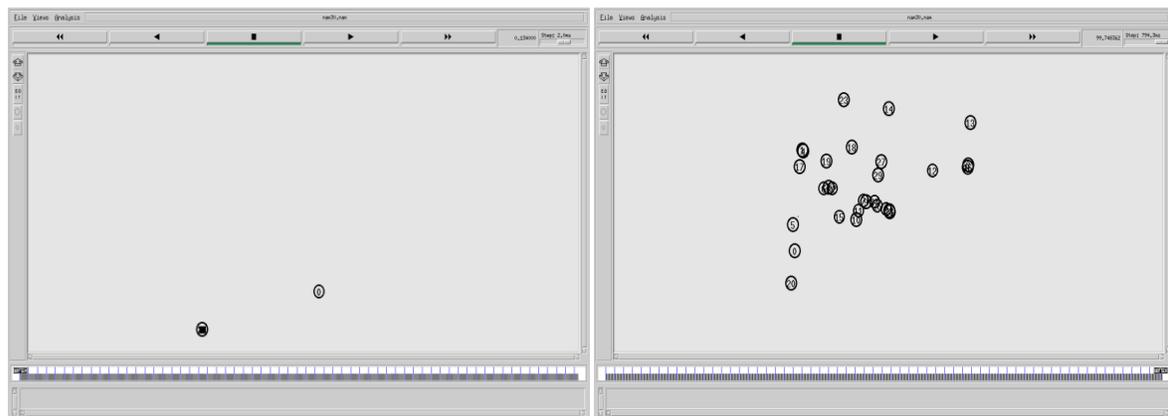
```

Proses ini akan menghasilkan dua *file* baru yaitu *trace file* (.tr) dan *nam file* (.nam). sedangkan proses untuk menampilkan animasi simulasi jaringan menggunakan nam adalah seperti pada *script* berikut, sedangkan tampilannya seperti pada Gambar 4.4

```

$ nam nam-30.nam

```



(a)

(b)

Gambar 4.4 Animasi simulasi dengan NAM, (a) pergerakan *node* detik awal; (b) pergerakan *node* detik terakhir

Pada gambar 4.4 merupakan tampilan animasi menggunakan contoh 30 *node*, pada gambar (a) menampilkan pergerakan *node* pada detik-detik awal simulasi yang dimana pada gambar tersebut menunjukkan bahwa pada detik awal *node* tidak muncul secara bersamaan melainkan akan muncul

satu persatu seiring dengan berjalannya waktu simulasi, kemunculan *node* tersebut digenerate secara *random* oleh sistem NS2 yang dilakukan pada saat proses perancangan skenario simulasi, pola pergerakan *node* pada simulasi tersebut juga digenerate secara *random* oleh sistem NS2. Sedangkan pada gambar (b) memperlihatkan pergerakan *node* pada detik akhir simulasi, gambar tersebut menunjukkan bawa keseluruhan *node* telah muncul.

### 4.3 Analisis Hasil Simulasi Menggunakan AWK

Untuk mengetahui unjuk kinerja jaringan pada tahap ini akan dilakukan analisis *file trace* menggunakan suatu bahasa pemrograman bernama AWK. Parameter yang digunakan untuk mengukur kinerja topologi jaringan tersebut adalah *packet loss*, *throughput*, dan *end to end delay*. Berikut adalah perintah analisis *file trace* menggunakan AWK

```
awk -f qos.awk trace-30.tr
```

### 4.4 Hasil Uji Coba

Pada sub bab ini akan ditampilkan bagaimana hasil uji coba simulasi yang telah dilakukan. Hasil ini merupakan analisis terhadap dampak perubahan lingkungan propagasi terhadap kinerja protokol *routing* reaktif, proaktif, dan *hybrid*. Beberapa jenis propagasi yang digunakan antara lain adalah *FreeSpace*, *TwoRayGround*, *Shadowing*, dan *Nakagami*. Analisis ini juga melibatkan perubahan jumlah *node* dalam lingkungan simulasi, secara berurutan *node* yang akan dimasukkan adalah 30, 50, dan 70. Sedangkan parameter untuk menguji kinerja jaringan adalah *packet loss*, *throughput*, dan *end to end delay*. Berikut adalah data yang diperoleh setelah dilakukan uji coba simulasi.

#### 4.4.1 Unjuk Kinerja Berdasarkan Nilai *Packet Loss*

Uji coba yang pertama adalah mencari nilai persentase rata-rata paket yang hilang selama rentang waktu simulasi. Tabel 4.1 memperlihatkan hasil dari unjuk kinerja perubahan lingkungan propagasi terhadap protokol *routing* reaktif, proaktif, serta *hybrid*, yang juga melibatkan perubahan jumlah *node* yaitu 30, 50, dan 70 berdasarkan nilai persentase *packet loss*. Nilai berikut merupakan hasil percobaan yang dilakukan sebanyak 10 kali percobaan, kemudian diambil nilai rata-ratanya.

Tabel 4.1 Kinerja berdasarkan persentase *packet loss*

Jenis Popagasi	<i>packet loss (%)</i>								
	AOMDV			OLSR			ZRP		
	30	50	70	30	50	70	30	50	70
<i>FreeSpace</i>	0.03	0	0	0	0.201	0	0.077	11.904	48.853
<i>TwoRayGround</i>	6.706	20.761	19.457	25.291	39.564	28.749	2.865	26.948	34.14
<i>Shadowing</i>	0.028	0.075	0.141	0.242	0.076	0.104	0.242	13.721	50.008
<i>Nakagami</i>	0	0	0	0.11	0.115	0	0.11	1.312	46.1

Pada Tabel 4.1 merupakan hasil rata-rata nilai pengujian *packet loss*, didapatkan beberapa hasil terbaik atau hasil minimum yang menyatakan bahwa rata-rata tidak terdapat paket yang hilang atau bernilai 0 selama rentan waktu simulasi. Hasil terbaik ditunjukkan oleh Protokol *routing* AOMDV dengan propagasi *Nakagami* dan *FreeSpace* karena rata-rata nilai *packet loss* adalah 0% untuk semua jumlah variasi *node*, sedangkan hasil terburuk ditunjukkan oleh propagasi *TwoRayGround* pada semua jenis protokol *routing*.

#### 4.4.2 Unjuk Kinerja Berdasarkan Nilai *Throughput*

Uji coba yang kedua adalah mencari nilai rata-rata paket yang diterima yang berhasil diterima oleh *node* penerima pada setiap detikanya (*throughput*). Tabel 4.2 memperlihatkan hasil dari unjuk kinerja perubahan lingkungan propagasi terhadap protokol *routing* reaktif, proaktif, serta *hybrid*, yang juga melibatkan perubahan jumlah *node* yaitu 30, 50, dan 70 berdasarkan nilai *throughput*. Nilai berikut merupakan hasil percobaan yang dilakukan sebanyak 10 kali percobaan, kemudian diambil nilai rata-ratanya.

Tabel 4.2 Kinerja berdasarkan *throughput*

Jenis Propagasi	<i>throughput (kbps)</i>								
	AOMDV			OLSR			ZRP		
	30	50	70	30	50	70	30	50	70
<i>FreeSpace</i>	29.45	30.191	29.815	29.778	29.9	29.457	29.595	25.47	14.262
<i>TwoRayGround</i>	27.289	23.832	23.509	22.187	18.519	20.79	29.181	22.6356	19.016
<i>Shadowing</i>	29.04	29.618	29.498	29.529	29.779	29.581	29.565	25.304	14.782
<i>Nakagami</i>	29.315	29.814	29.999	29.772	29.77	29.69	29.722	29.198	15.511

Pada Tabel 4.2 merupakan nilai rata-rata hasil pengujian *throughput*, yang apabila diambil rata-rata pada *node* 30, 50, dan 70 maka hasil terbaik terdapat pada protokol *routing* AOMDV dengan propagasi *FreeSpace* dengan nilai rata-ratanya adalah sebesar 29.818 kbps, dan sedangkan hasil terburuk ditunjukkan oleh protokol *routing* OLSR propagasi *TwoRayGround* dengan nilai rata-rata adalah sebesar 20.498 kbps.

#### 4.4.3 Unjuk Kinerja Berdasarkan Nilai *End To End Delay*

Uji coba yang ketiga adalah mencari nilai rata-rata selang waktu pengiriman paket data dari *node* sumber sampai ke *node* tujuan (*end to end delay*). Tabel 4.3 memperlihatkan hasil dari unjuk kinerja perubahan lingkungan propagasi terhadap protokol *routing* reaktif, proaktif, serta *hybrid*, yang juga melibatkan perubahan jumlah *node* yaitu 30, 50, dan 70 berdasarkan nilai *end to end delay*. Nilai berikut merupakan hasil percobaan yang dilakukan sebanyak 10 kali percobaan, kemudian diambil nilai rata-ratanya.

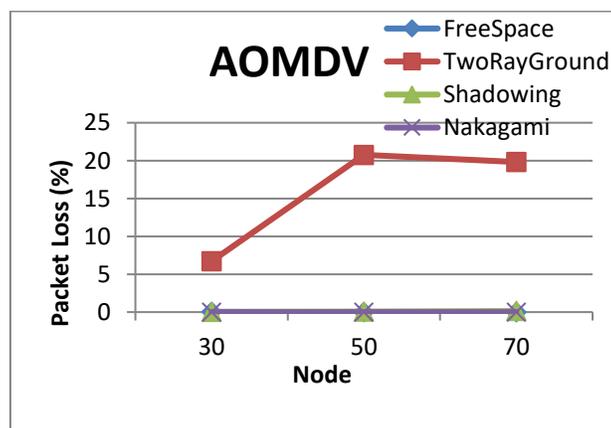
Tabel 4.3 Kinerja berdasarkan *end to end delay*

Jenis Propagasi	<i>end to end delay (m/s)</i>								
	AOMDV			OLSR			ZRP		
	30	50	70	30	50	70	30	50	70
<i>FreeSpace</i>	6.3223	6.2399	6.1960	5.9856	6.0799	6.2344	24.306	882.71	2609.4
<i>TwoRayGround</i>	8.1754	14.119	12.817	7.6486	10.235	12.788	11.708	75.968	112.56
<i>Shadowing</i>	6.7918	7.242	9.2633	32.239	6.8569	6.8163	32.493	899.30	2115.3
<i>Nakagami</i>	6.1313	6.2550	6.9040	16.602	6.0961	6.1549	16.602	161.12	2139.0

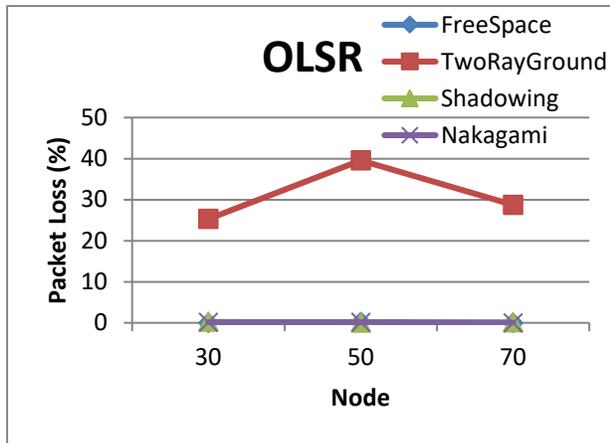
Pada Tabel 4.3 merupakan nilai rata-rata hasil pengujian *end to end delay*, didapatkan hasil terbaik atau minimum yaitu dengan hasil rata-rata 5.9856 pada protokol *routing* OLSR propagasi *FreeSpace* dengan jumlah *node* 30.

#### 4.5 Analisis Hasil Uji Coba

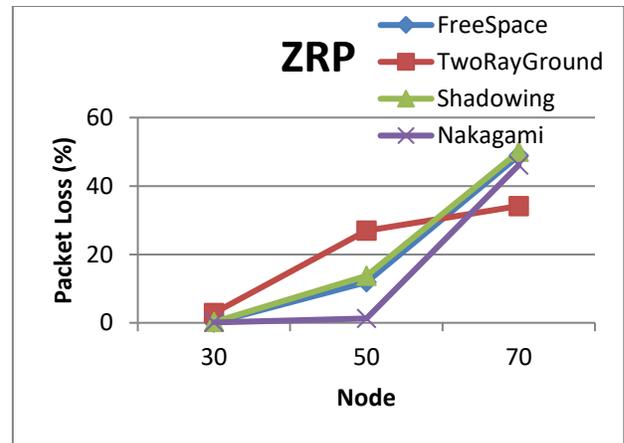
Gambar 4.5 merupakan hasil rata-rata *packet loss* sesuai dengan hasil uji coba skenario pada Tabel 4.1



(a)



(b)

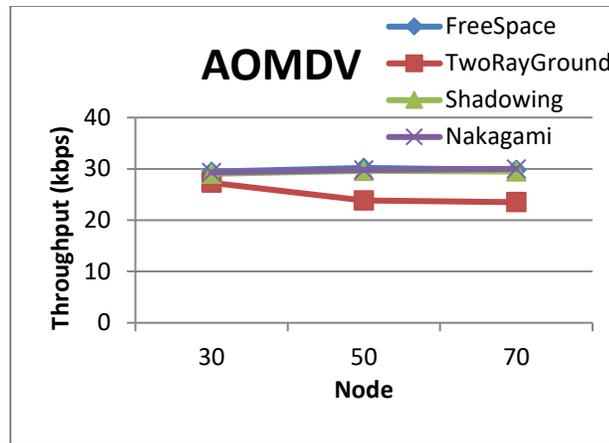


(c)

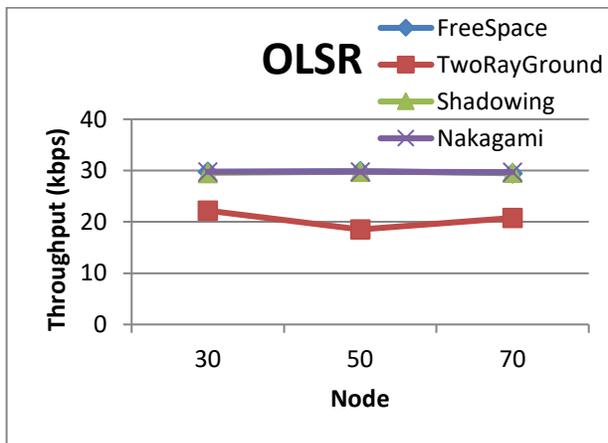
Gambar 4.5 Perbandingan nilai rata-rata *packet loss*. (a) grafik perbandingan nilai *packet loss* protokol *routing* AOMDV; (b) grafik perbandingan nilai *packet loss* protokol *routing* OLSR; (c) grafik perbandingan nilai *packet loss* protokol *routing* ZRP;

Pada gambar 4.5 grafik (a) menunjukkan bahwa apabila menggunakan protokol *routing* AOMDV, propagasi *TwoRayGround* menghasilkan nilai *packet loss* yang lebih besar untuk semua variasi jumlah *node*, nilai *packet loss* terbesar terjadi pada variasi jumlah 50 *node* yaitu 20.761%. Sedang untuk propagasi *FreeSpace*, *Shadowing*, dan *Nakagami* nilai *packet loss* hampir sama dan tidak mengalami perubahan signifikan untuk semua variasi jumlah *node*. Grafik (b) menunjukkan bahwa apabila menggunakan protokol *routing* OLSR, propagasi *TwoRayGround* menghasilkan nilai *packet loss* yang lebih besar untuk semua variasi jumlah *node*, nilai *packet loss* terbesar terjadi pada variasi jumlah 50 *node* yaitu 39.564%. Sedang untuk propgasi *FreeSpace*, *Shadowing*, dan *Nakagami* nilai *packet loss* hampir sama dan tidak mengalami perubahan signifikan untuk semua variasi jumlah *node*. Grafik (c) menunjukkan bahwa apabila menggunakan protokol *routing* ZRP, untk semua jenis propagasi nilai *packet loss* akan bertambah seiring dengan bertambahnya jumlah *node*, akan tetapi hasil terbesar tunjukan oleh propagasi *Shadowing* pada variasi 70 *node* dengan nilai 50.008%, sedangkan hasil terkecil ditunjukan oleh propagasi *FreeSpace* pada variasi jumlah 30 *node* dengan nilai 0.077%.

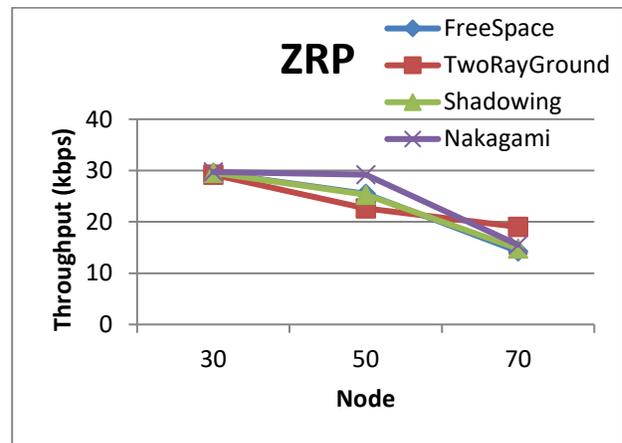
Gambar 4.6 merupakan hasil rata-rata *throughput* sesuai dengan hasil uji coba skenario pada Tabel 4.2



(a)



(b)



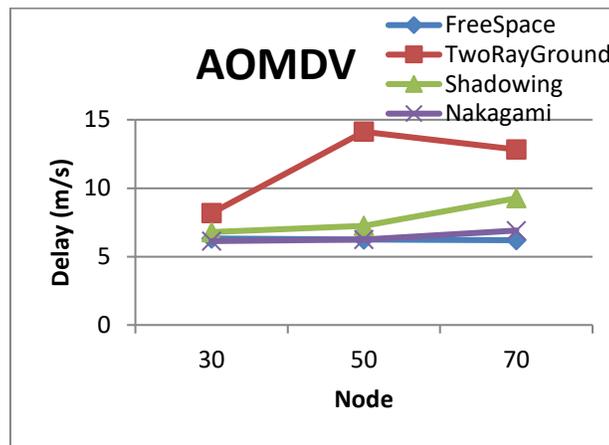
(c)

Gambar 4.6 Perbandingan nilai rata-rata *throughput*. (a) grafik perbandingan nilai *throughput* protokol *routing* AOMDV; (b) grafik perbandingan nilai *throughput* protokol *routing* OLSR; (c) grafik perbandingan nilai *throughput* protokol *routing* ZRP;

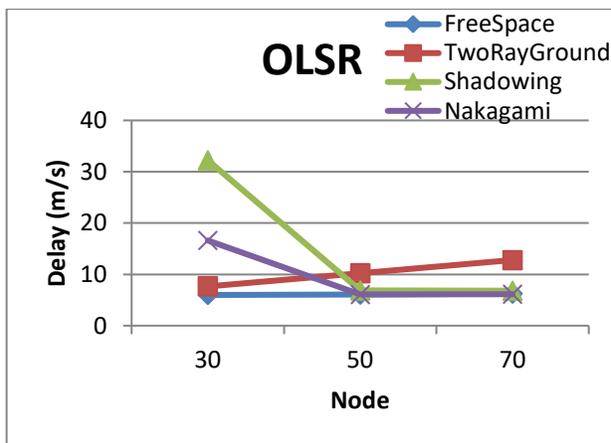
Pada Gambar 4.6 grafik (a) menunjukkan bahwa apabila menggunakan protokol *routing* AOMDV, propagasi *TwoRayGround* memiliki nilai *throughput* yang lebih rendah untuk semua variasi jumlah *node*, nilai paling rendah ditunjukkan pada variasi jumlah 70 *node* dengan nilai 23.509 kbps. Sedangkan untuk propagasi *FreeSpace*, *Shadowing*, dan *Nakagami* memiliki nilai *throughput* yang hampir sama dan tidak mengalami perubahan signifikan untuk semua variasi jumlah *node*, akan tetapi hasil tertinggi ditunjukkan oleh propagasi *FreeSpace* pada variasi jumlah 50 *node* dengan nilai 30.191 kbps. Grafik (b) menunjukkan bahwa apabila menggunakan protokol *routing* OLSR, propagasi *TwoRayGround* memiliki nilai *throughput* yang lebih rendah untuk semua variasi jumlah *node*, nilai paling rendah ditunjukkan pada variasi jumlah 50 *node* dengan nilai 18.519 kbps. Sedangkan untuk propagasi *FreeSpace*, *Shadowing*, dan *Nakagami* memiliki nilai *throughput* yang hampir sama dan tidak

mengalami perubahan signifikan untuk semua variasi jumlah *node*, akan tetapi hasil tertinggi ditunjukkan oleh propagasi *FreeSpace* pada variasi jumlah 50 *node* dengan nilai 29.9 kbps. Grafik (c) menunjukkan bahwa apabila menggunakan protokol *routing* ZRP, untuk semua jenis propagasi nilai *throughput* akan berkurang seiring dengan bertambahnya jumlah *node*, akan tetapi hasil terbesar ditunjukkan oleh propagasi *Nakagami* dengan nilai 29.722 kbps, sedangkan nilai terkecil ditunjukkan oleh propagasi *FreeSpace* dengan nilai 14.262 kbps.

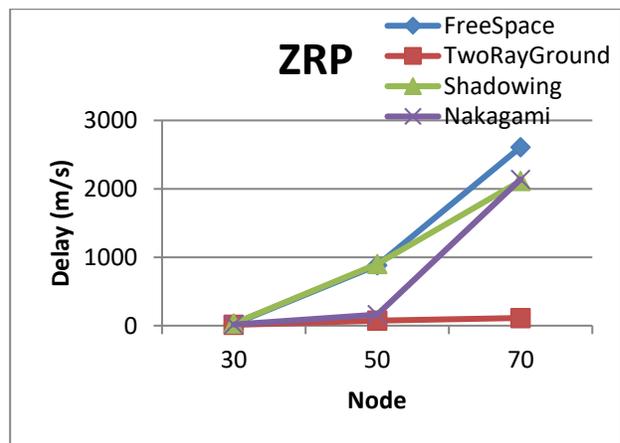
Gambar 4.7 merupakan hasil rata-rata *end to end delay* sesuai dengan hasil uji coba skenario pada Tabel 4.3



(a)



(a)



(b)

Gambar 4.7 Perbandingan nilai rata-rata *end to end delay*. (a) grafik perbandingan nilai *end to end delay* protokol *routing* AOMDV; (b) grafik perbandingan nilai *end to end delay* protokol *routing* OLSR; (c) grafik perbandingan nilai *end to end delay* protokol *routing* ZRP;

Pada Gambar 4.7 grafik (a) menunjukkan bahwa apabila menggunakan protokol *routing* AOMDV, propagasi *TwoRayGround* memiliki nilai *end to end delay* yang paling besar pada semua

variasi jumlah *node*, nilai terbesar ditunjukkan pada variasi jumlah 50 *node* yaitu 14.119 m/s. Sedangkan nilai terkecil ditunjukkan oleh propagasi *Nakagami* pada variasi jumlah 30 *node* dengan nilai 6.1313 m/s. Grafik (b) menunjukkan bahwa apabila menggunakan protokol *routing* OLSR, propagasi *Shadowing* memiliki nilai *end to end delay* yang paling besar, yaitu pada variasi jumlah 30 *node* dengan nilai 32.239 m/s, sedangkan nilai terkecil ditunjukkan propagasi *FreeSpace* pada variasi jumlah 30 *node* yaitu dengan nilai 5.9856 m/s. Grafik (c) menunjukkan bahwa apabila menggunakan protokol *routing* ZRP, untuk semua jenis propagasi nilai *end to end delay* mengalami peningkatan seiring dengan bertambahnya jumlah *node*, akan tetapi nilai terbesar ditunjukkan oleh propagasi *FreeSpace* pada variasi jumlah 70 *node* dengan nilai 2609.4 m/s, sedangkan nilai terkecil ditunjukkan oleh propagasi *TwoRayGround* dengan nilai 11.708 m/s pada variasi jumlah 30 *node*.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah dilakukan proses analisis terhadap hasil simulasi jaringan VANET terhadap protokol *routing* AOMDV, OLSR, dan ZRP dengan menggunakan peta jalan kota Bima, maka didapat kesimpulan sebagai berikut:

1. Berdasarkan *packet loss* protokol *routing* AOMDV dan OLSR mendapatkan nilai rata-rata sebesar 0% untuk semua variasi jumlah *node* pada propagasi *Nakagami* untuk AOMDV dan propagasi *FreeSpace* untuk OLSR, sedangkan ZRP mendapatkan nilai minimum sebesar 0.077% pada propagasi *FreeSpace node 30*.
2. Nilai rata-rata *throughput* protokol *routing* AOMDV terbaik yaitu menggunakan jenis propagasi *Free Space* yaitu 30.191 kbps pada *node 50*. Pada protokol *routing* OLSR hasil rata-rata terbaik kembali pada jenis propagasi *Free Space* dengan 29.9 kbps pada *node 50*. Sedangkan pada protokol *routing* ZRP propagasi *Nakagami* unggul dengan nilai 29.722 kbps pada *node 30*.
3. Nilai rata-rata *delay* terbaik untuk masing-masing protokol *routing* AOMDV, OLSR, dan ZRP secara berurutan adalah 6.1313 m/s pada jumlah *node 30* propagasi *Nakagami*, 5.9856 m/s pada jumlah *node 30* propagasi *FreeSpace*, 11.708 m/s pada jumlah *node 30* propagasi *TwoRayGround*.
4. Berdasarkan keseluruhan hasil percobaan simulasi yang digunakan pada peta jalan kota Bima protokol *routing* paling cocok diterapkan adalah AOMDV sedangkan untuk jenis propagasinya adalah *Nakagami*.

#### 5.2 Saran

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

1. Melakukan penelitian yang serupa dengan mengimplementasikan algoritma tambahan serta menggunakan simulator jenis lain seperti OMNet++ dan NS3.
2. Melakukan penelitian menggunakan protokol *routing* selain AOMDV, OLSR, dan ZRP.

## DAFTAR PUSTAKA

- [1] V. D. Khairnar and K. Kotecha, "Propagation Models For V2V Communication In Vehicular Ad-Hoc Networks," *J. Theor. Appl. Inf. Technol.*, vol. 61, no. 3, pp. 686–695, 2014.
- [2] S. Dubey, A. Nema, and A. Verma, "AODV and AOMDV Routing Protocol with 801 . 11p in VANET," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 6, no. 6, pp. 136–140, 2016.
- [3] M. D. Agustya, R. Anggoro, and H. Studiawan, "Analisis Performa Proactive Routing Protocol DSDV dan OLSR pada Vehicular Ad hoc Network ( VANET ) Menggunakan NS-3," *Institut Teknologi Sepuluh November, Program Studi Teknik Informatika*, 2015.
- [4] A. Setiabudi, A. A. Pratiwi, Ardiansyah, D. Perdana, and R. F. Sari, "Analisis Performa Proactive Routing Protocol DSDV dan OLSR pada Vehicular Ad hoc Network ( VANET ) Menggunakan NS-3," *Reg. 10 Symp.*, 2016.
- [5] K. A. Hafeez, L. Zhao, Z. Liao, and B. N. W. Ma, "The optimal radio propagation model in VANET," *4th Int. Conf. Syst. Networks Commun. ICSNC 2009*, pp. 6–11, 2009.
- [6] D. S. Ramadhani and R. Anggoro, "Studi Perbandingan Kinerja Model Transmisi TwoRayGround dan Nakagami pada OLSR di Lingkungan MANET Menggunakan NS-2," *J. Tek. ITS*, vol. 5, no. 2, 2016.
- [7] R. C. Poonia and V. Singh, "Performance Evaluation of Radio Propagation Model For Vehicular Ad Hoc Networks Using VanetMobisim And NS-2," *Int. J. Distrib. Parallel Syst.*, vol. 3, no. 4, pp. 145–155, 2012.
- [8] B. Makodia and D. Marsonia, "Effect of Radio-Wave Propagation Model On Network Performance with AODV Routing Protocol For IL-VANET," *Int. Arch. Appl. Sci. Technol.*, vol. 5, pp. 13–18, 2014.
- [9] Y. A. Pranata, I. Fibriani, and S. B. Utomo, "Analisis Optimasi Kinerja Quality of Service Pada Layanan Komunikasi Data Menggunakan NS-2 di PT . PLN ( Persero ) Jember," vol. 20, no. 2, pp. 149–156, 2016.
- [10] R. Kumar and M. Dave, "A Comparative Study of Various Routing Protocols in VANET," *IJCSI Int. J. Comput. Sci. Issues*, vol. 8, no. 4, pp. 643–648, 2011.

- [11] R. Hadiwiryanto, P. H. Trisnawan, and K. Amron, "Implementasi Protokol Geographic Source Routing ( GSR ) Pada Vehicular Ad-Hoc Network ( VANET ) untuk Komunikasi Kendaraan Dengan Road Side Unit ( RSU )," *J. Penegembangan Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 12, pp. 7007–7016, 2018.
- [12] K. Kamarullah, Endroyono, and Wirawan, "Optimasi Cross Layer Untuk Protokol Dynamic Source Routing Pada Komunikasi Antar Kendaraan Berbasis Vehicular Ad-Hoc Networks ( VANETs )," *J. Tek. ITS*, vol. 6, no. 2, pp. 443–448, 2017.
- [13] R. Wulandari, "Analisis QOS ( Quality Of Service ) Pada Jaringan Internet ( Studi Kasus : UPT Loka Uji Teknik Penambangan Jampang Kulon – LIPI )," *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 2, pp. 162–172, 2016.
- [14] G. A. Romanta, "Simulasi Vehicular Ad Hoc Network ( VANET ) Menggunakan Openstreetmap Dengan Simulation Of Urban Mobility ( SUMO ) Dan Omnet ++," 2017.
- [15] P. K. Singh, "Comparative Study of Radio Propagation and Mobility Models in Vehicular Adhoc Network," *Int. J. Comput. Appl. (0975)*, vol. 16, no. 8, pp. 37–42, 2011.
- [16] F. Mapa, S. Djanali, and A. M. S., "Optimasi OLSR Routing Protocol Pada Jaringan Wireless Mesh Dengan Adaptive Refreshing Time Interval Dan Enhance Multi Point Relay Selecting Algorithm," vol. 12, no. 1, pp. 44–49, 2014.