

SIMULASI DAN ANALISIS PROTOKOL 802.11b PADA VEHICULAR AD-HOC NETWORK (VANET) UNTUK KOMUNIKASI ANTAR KENDARAAN

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Diko Firmandiansyah Putra

NIM: 115060801111052



PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2015

PENGESAHAN

SIMULASI DAN ANALISIS PROTOKOL 802.11b PADA VEHICULAR AD-HOC
NETWORK (VANET) UNTUK KOMUNIKASI ANTAR KENDARAAN

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Diko Firmandiansyah Putra
NIM: 115060801111052

Skripsi ini telah diuji dan dinyatakan lulus pada
29 Oktober 2015

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Sabriansyah Rizgika A, S.T, M. Eng.
NIP: 19820809 201212 1 004

Adhitya Bhawiyuga, S.Kom, M.S.
NIK: 201405 890720 1 1 001

Mengetahui
Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, M.T.
NIP: 19670801 199203 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 29 Oktober 2015



Diko Firmandiansyah Putra

NIM: 1115060801111052

KATA PENGANTAR

Puji syukur penulis ucapkan ke hadirat Allah SWT yang telah melimpahkan segala Rahmat, Karunia, dan Hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul : **“Simulasi dan Analisis Protokol 802.11b Pada Vehicular Ad-Hoc Network (VANET) Untuk Komunikasi Antar Kendaraan”**.

Skripsi ini diajukan sebagai syarat ujian seminar skripsi dalam rangka untuk memperoleh gelar Sarjana Komputer di Program Studi Teknik Informatika dan Ilmu Komputer Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya Malang, atas terselesaikannya skripsi ini, penulis mengucapkan terima kasih kepada :

1. Ibunda Eko Dyah Fillyantarie, Ayahanda Eddy Saptono dan Kakak Dike Yanuarista yang telah banyak memberikan bantuan moril, material, arahan dan selalu mendoakan keberhasilan dan keselamatan selama menempuh pendidikan.
2. Alm. Kakek Moedjiatno dan Almh. Nenek Liliek Sirtufillaili yang telah memberikan banyak tauladan hidup sejak saya masih kecil hingga dewasa.
3. Bapak Sabriansyah Rizqika A, S.T, M.Eng. dan Bapak Adhitya Bhawiyuga, S.Kom, M.S. selaku Dosen Pembimbing Skripsi yang telah memberikan bimbingan dalam proses skripsi ini.
4. Dzikrillah Rizqi Amalia yang selalu memberikan dukungan untuk menyelesaikan laporan skripsi ini.
5. Iqbal, Julian dan Happy yang telah bersama-sama mengerjakan tugas akhir ini.
6. Bogi, Kidi, Tomi, Wiki, Ganda, Johan, Krus, Ulum, Imam Buchory dan Radit yang sudah bekerjasama dalam perkuliahan sejak awal semester.
7. Mas Putra, Mas Dika dan Mbak Ayu yang telah banyak membantu saya dalam hal perkuliahan.
8. Rekan-rekan Hefotris, SPFC Family dan TIF-D yang telah membantu dalam penyelesaian laporan skripsi ini.
9. Berbagai pihak yang telah memberikan bantuan dan dorongan serta berbagi pengalaman pada proses penyusunan skripsi ini.

Penulis menyadari bahwa skripsi ini masih memiliki kekurangan dan masih belum sempurna, karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Maka, saran dan kritik yang membangun dari semua pihak sangat diharapkan demi penyempurnaan selanjutnya. Semoga skripsi ini dapat bermanfaat dan berguna bagi semua pihak, baik penulis maupun pembaca, dan semoga Allah SWT meridhoi dan dicatat sebagai amalan ibadah, Amin.

Malang, 29 Oktober 2015

Penulis

dikofirmandiansyah@gmail.com

ABSTRAK

Saat ini komunikasi antara kendaraan dan infrastruktur disebut dengan *Vehicular Ad-Hoc Network* (VANET) merupakan sebuah topik hangat penelitian pada jaringan. Salah satu pokok permasalahan pada vanet adalah *mobility* dari node kendaraan yang berpindah sangat cepat. Teknologi untuk vanet saat ini adalah protokol komunikasi yang menggunakan IEEE 802.11p sebagai *physical layer*. Namun, masih terdapat beberapa kelemahan pada protokol tersebut yaitu *device* yang digunakan masih jarang beredar di pasaran. Pada penelitian ini menggunakan standar protokol komunikasi IEEE 802.11b yang memiliki *data rate* sebesar 11 Mbps dengan *channel* frekuensi 2.4 GHz. Untuk menganalisa jaringan vanet dapat diterapkan pada komunikasi antar kendaraan.

Pada penelitian ini menguji pengaruh parameter kepadatan lalu lintas, besar pengiriman data dan kecepatan kendaraan terhadap kinerja dari komunikasi vanet. Pada skenario kepadatan lalu lintas memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 95942.9256 bps, 347.1401 ms dan 0.04 %. Pada skenario besar pengiriman data memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 56509.3635 bps, 270.256 ms dan 0 %. Sedangkan pada skenario kecepatan kendaraan memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 80701.04 bps, 386.26 ms dan 0 %.

Kata kunci: VANET, OMNeT++, SUMO, INET, 802.11b.

ABSTRACT

Currently, the communication between vehicles and the infrastructure called Vehicular Ad-Hoc Network (VANET) is favored research topic on the area of networking. One of the principal problems in the VANET is the high mobility from vehicles node. Technology for VANET today is a communication protocol which uses IEEE 802.11p as the physical layer. However, there are still some weaknesses in the protocol that the device is still rare in the market. In this study, we use IEEE 802.11b as standard communication protocol which has a data rate of 11 Mbps with channel frequency of 2.4 GHz. VANET can be applied for the communication between vehicles.

In this study, we examine the impact of traffic density, transmitted data length and vehicle speed of VANET communications to the performance of VANET in term of throughput, End-to-End Delay and packet loss. In the scenario with variated traffic density, the throughput, End-to-End Delay and packet loss are 95942.9256 bps, 347.1401 ms and 0.04 %, respectively. In the scenario with variated transmitted data length, the throughput, End-to-End Delay and packet loss are 56509.3635 bps, 270.256 ms and 0 %, respectively. In the scenario with variated vehicle speed, the throughput, End-to-End Delay and packet loss are 80701.04 bps, 386.26 ms and 0 %, respectively.

Keywords: VANET, OMNeT++, SUMO, INET, 802.11b.

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	2
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Vehicular Ad-Hoc Network (VANET)	5
2.3 OMNeT++	6
2.4 Simulation of Urban Mobility (SUMO).....	7
2.5 Standar IEEE 802.11b.....	8
2.5.1 Frekuensi IEEE 802.11b	9
2.5.2 Modulasi IEEE 802.11b.....	9
2.6 INET Framework.....	11
2.7 End-to-End Delay	11
2.8 Throughput	12
2.9 Packet Loss.....	13
BAB 3 METODOLOGI	14
3.1 Metodologi Penelitian	14
3.1.1 Studi Literatur	14
3.1.2 Perancangan Simulasi	15

3.1.3 Implementasi	16
3.1.4 Pengujian dan Analisis.....	16
3.1.5 Pengambilan Kesimpulan	17
BAB 4 HASIL	18
4.1 Rancangan Implementasi	18
4.1.1 Perancangan Peta dan Keadaan lalu Lintas pada SUMO	18
4.1.2 Perancangan Sistem Jaringan Pada Node Kendaraan	23
4.1.3 Konfigurasi Standar IEEE 802.11b	25
4.2 Pengujian	25
4.2.1 Pengujian Terhadap Kepadatan	25
4.2.2 Pengujian Terhadap Besar Data	27
4.2.3 Pengujian Terhadap Kecepatan	28
BAB 5 PEMBAHASAN	30
5.1 Hasil.....	30
5.1.1 Pengujian Terhadap Parameter Kepadatan Kendaraan	30
5.1.2 Pengujian Terhadap Parameter Besar Data	32
5.1.3 Pengujian Terhadap Parameter Kecepatan	33
5.2 Analisis	35
5.2.1 Analisis Pengujian Terhadap Pengaruh Kepadatan	35
5.2.2 Analisis Pengujian Terhadap Pengaruh Besar Data	36
5.2.3 Analisis Pengujian Terhadap Pengaruh Kecepatan.....	36
BAB 6 PENUTUP	37
6.1 Kesimpulan.....	37
6.2 Saran	37
DAFTAR PUSTAKA.....	38

DAFTAR TABEL

Tabel 3.1 List Skenario Pengujian.....	17
Tabel 4.1 Parameter 802.11b.....	25



DAFTAR GAMBAR

Gambar 2.1 Channel Frekuensi 2.4GHz	8
Gambar 2.2 Skema Lapisan Fisik 802.11b.....	10
Gambar 3.1 Flowchart Metode Penelitian.....	14
Gambar 3.2 Alur Kerja Simulator	15
Gambar 4.1 Tampilan Aplikasi Java OpenStreetMap (JOSM)	18
Gambar 4.2 Peta yang akan digunakan	19
Gambar 4.3 Konversi file OSM	20
Gambar 4.4 Gambar file berekstensi .net.xml, .rou.xml, .sumo.cfg dan .osm	21
Gambar 4.5 Konfigurasi untuk Menampilkan Nilai Throughput.....	22
Gambar 4.6 Import Traci Car dan IPv4NetworkConfigurator	22
Gambar 4.7 Konfigurasi Menampilkan dan Record Hasil Delay dan Throughput	23
Gambar 4.8 Konfigurasi Sistem Jaringan Kendaraan	24
Gambar 4.9 Peta untuk Pengujian	26
Gambar 4.10 Konfigurasi untuk 10 jumlah kendaraan	26
Gambar 4.11 Konfigurasi untuk 20 jumlah kendaraan	26
Gambar 4.12 Konfigurasi untuk 30 jumlah kendaraan	26
Gambar 4.13 Konfigurasi untuk 40 jumlah kendaraan	26
Gambar 4.14 Konfigurasi untuk 50 jumlah kendaraan	27
Gambar 4.15 Map Jalan Lurus Dua Arah Berlawanan	27
Gambar 4.16 Konfigurasi pengiriman pesan.....	27
Gambar 4.17 File Ned untuk Kendaraan.....	28
Gambar 4.18 Konfigurasi pada omnetpp.ini.....	28
Gambar 4.19 Konfigurasi kecepatan pada SUMO	28
Gambar 5.1 Hasil Throughput pengujian kepadatan	30
Gambar 5.2 End-to-End Delay pada pengujian kepadatan.....	31
Gambar 5.3 Packet loss dari pengujian kepadatan.....	31
Gambar 5.4 Hasil Throughput pada Pengujian Besar Data.....	32
Gambar 5.5 Delay pada pengujian besar data.....	33
Gambar 5.6 Hasil Packet loss pada Pengujian Besar Data	33
Gambar 5.7 Hasil Troughput pada pengujian kecepatan kendaraan	34
Gambar 5.8 Hasil <i>End-to-End Delay</i> pada pengujian kecepatan kendaraan	34

Gambar 5.9 Hasil Packet loss pada pengujian kecepatan kendaraan 35



BAB 1 PENDAHULUAN

1.1 Latar belakang

Saat ini, kendaraan telah menjadi kebutuhan pokok bagi masyarakat luas. Banyaknya manfaat yang didapatkan dari fungsi kendaraan mengakibatkan bertambahnya jumlah pengguna kendaraan. Namun, peningkatan kendaraan dan jumlah penggunaannya tidak sebanding dengan kualitas infrastruktur, sehingga dapat menyebabkan jumlah resiko dan efek kecelakaan di suatu daerah akan bertambah tinggi pula. Beberapa tindakan sudah dilakukan untuk mengurangi efek suatu kecelakaan contohnya dengan memasang *safety belt* dan *airbag* di dalam kendaraan, akan tetapi langkah tersebut hanya sebatas untuk mengurangi efek pada kecelakaan yang terjadi tanpa dapat mencegah terjadinya suatu kecelakaan.

Untuk membenahi permasalahan tersebut, telah berkembang sebuah konsep baru yang memungkinkan komunikasi antar kendaraan (*inter vehicle*) dan komunikasi antara kendaraan dengan infrastruktur di sekitar jalan (*vehicle to infrastructure*). Konsep jaringan *wireless* ini dikenal sebagai *Vehicular Ad-Hoc Network* (VANET) yang dapat digunakan untuk mencegah kecelakaan dan menyediakan aplikasi-aplikasi transportasi untuk keperluan keamanan hingga aplikasi untuk kenyamanan pengemudi. Melalui komunikasi antar kendaraan (*inter vehicle*), kendaraan di suatu jalan dapat saling bertukar data sesuai dengan informasi yang dibutuhkan. Untuk mencegah kecelakaan, aplikasi pada VANET akan memonitor kendaraan satu dengan kendaraan lainnya dengan mengirimkan *safety status message* dan *warning message* (dikendalikan oleh event/kejadian). *Warning message* mengandung informasi-informasi terkait keamanan, sedangkan *safety status message* dikirim secara periodik ke semua kendaraan yang berada dalam jangkauannya yang pesannya berisikan status kendaraan pengirim seperti kecepatan, akselerasi, arah dan posisi. Maka dari itu, dengan adanya aplikasi kedua *message* tersebut, *Vehicular Ad-Hoc Network* (VANET) dapat mencegah terjadinya kecelakaan.

Protokol standar yang digunakan pada VANET adalah 802.11p, akan tetapi masih terdapat beberapa kelemahan pada protokol tersebut yaitu *device* yang digunakan masih jarang beredar di pasaran. Untuk mengatasi kekurangan tersebut, akan lebih baik jika menggunakan protokol 802.11b dimana *device* yang mendukung kinerja protokol ini telah banyak digunakan di pasaran. Hal ini tentu mendukung *requirement* pada *Vehicular Ad-Hoc Network* yang mengharuskan adanya ketersediaan pengguna supaya menghasilkan komunikasi *ad-hoc* yang dapat berjalan dengan baik. Untuk mendukung tujuan tersebut dapat dilakukan sebuah simulasi yang berguna mendapatkan hasil dan menguji performansi protokol 802.11b pada *Vehicular Ad-Hoc Network*.

Pada topik penelitian yang berjudul “Simulasi dan Analisis Protokol 802.11b Pada Vehicular Ad-Hoc Network (VANET) Untuk Komunikasi Antar Kendaraan” akan mensimulasikan dan menguji apakah protokol 802.11b dapat

digunakan pada *Vehicular Ad-Hoc Network* Dengan menggunakan OMNeT++ sebagai *network simulator* dan SUMO sebagai *traffic simulator* mampu mendapatkan parameter pengujian seperti *throughput*, *end-to-end delay* dan *packet loss* yang hasilnya dapat dianalisis dan diuji untuk mendapatkan parameter hasil yang diharapkan.

1.2 Rumusan masalah

Berdasarkan latar belakang di atas, dapat dirumuskan beberapa masalah sebagai berikut:

1. Bagaimana membangun infrastruktur jaringan VANET dengan protokol 802.11b pada simulator OMNeT++?
2. Bagaimana menerapkan simulasi dari penggunaan VANET terhadap simulasi keadaan lalu lintas dengan menggunakan SUMO?
3. Bagaimana hasil skenario variasi kepadatan kendaraan, besar pengiriman data dan kecepatan kendaraan terhadap parameter pengujian seperti *throughput*, *end-to-end delay* dan *packet loss*?

1.3 Tujuan

Penelitian ini bertujuan untuk menganalisis teknologi protokol 802.11b untuk jaringan VANET pada simulator OMNeT++ dan menerapkannya pada simulasi keadaan lalu lintas sesuai pengaturan skenario dengan menggunakan aplikasi SUMO sebagai *traffic simulator*.

1.4 Manfaat

Penulisan skripsi ini diharapkan mempunyai manfaat yang baik dan berguna bagi pembaca dan penulis. Adapun manfaat yang diharapkan adalah sebagai berikut:

- a. Bagi penulis
 1. Sebagai media untuk pengimplementasian ilmu pengetahuan teknologi yang telah didapat selama mengikuti perkuliahan di Program Studi Informatika/Illmu Komputer Universitas Brawijaya.
 2. Mendapatkan pengetahuan dan wawasan tentang VANET.
- b. Bagi pembaca
 1. Mendapatkan wawasan akan simulasi dan pengujian pada VANET menggunakan protokol 802.11b.
 2. Sebagai bahan dan acuan dalam mengembangkan dan melanjutkan riset lanjutan yang memiliki keterkaitan dengan penelitian ini.

1.5 Batasan masalah

Agar permasalahan yang dirumuskan dapat lebih terfokus, maka penelitian ini dibatasi dalam hal:

1. Perancangan topologi jaringan VANET difokuskan pada simulasi kondisi keadaan lalu lintas sesuai dengan pengaturan skenario.

2. Penelitian ini mengabaikan aspek keamanan jaringan.
3. Komunikasi yang dibangun adalah komunikasi antar kendaraan (*Inter-vehicle Communication*).
4. Simulasi VANET diimplementasikan dengan standar jaringan *wireless* 802.11b.
5. Simulasi pada penelitian ini hanya mengukur kinerja protokol dalam jaringan sejumlah parameter yang digunakan seperti *throughput*, *end-to-end delay* dan *packet loss*.
6. Simulasi protokol 802.11b pada VANET dilakukan dengan menggunakan simulator OMNeT++.
7. Simulasi keadaan lalu lintas dilakukan dengan menggunakan simulator SUMO.

1.6 Sistematika pembahasan

Sistematika penulisan penelitian ditunjukkan untuk memberikan gambaran dan uraian dari penyusunan tugas akhir secara garis besar yang meliputi beberapa bab, sebagai berikut.

BAB I : Pendahuluan

Menjelaskan latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

BAB II : Landasan Kepustakaan

Menguraikan kajian pustaka dan dasar teori yang mendasari teknologi *vanet* dan protokol yang digunakan dalam simulasi.

BAB III : Metodologi

Menguraikan tentang metode dan langkah kerja yang terdiri dari studi literatur, analisis kebutuhan simulasi, perancangan system, implementasi dan analisis serta pengambilan kesimpulan.

BAB IV : Hasil

Menguraikan proses implementasi dari dasar teori yang telah dipelajari sesuai analisis dan perancangan sistem.

BAB V : Pembahasan

Memuat hasil pengujian dan analisis terhadap sistem yang telah direalisasikan

BAB VI : Penutup

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian program, serta saran-saran untuk pengembangan lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Dasar atau acuan yang berupa teori-teori atau temuan-temuan melalui hasil berbagai penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung yang valid. Salah satu data pendukung yang menurut peneliti perlu untuk dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Dalam hal ini, fokus pada penelitian terdahulu yang dijadikan acuan terkait dengan masalah tentang penerapan sistem komunikasi dengan menggunakan teknologi jaringan VANET.

Penelitian mengenai analisis performansi tentang VANET telah banyak dilakukan, salah satunya yang dilakukan oleh Harshad Phule pada penelitian yang berjudul "Public safety application for approaching emergency vehicle alert and accident reporting in VANETs using WAVE". Pada penelitian tersebut dibangun sebuah sistem VANET untuk mengurangi jumlah kecelakaan dengan cara menyebarkan paket yang berisi informasi bila sebuah kendaraan mengalami kecelakaan maka ia akan menyebarkan informasi tersebut kepada kendaraan yang ada disekitarnya guna menghindari kecelakaan yang lainnya. Penelitian tersebut menggunakan simulator OMNeT++ dan SUMO untuk menjalankan skenario. Protokol komunikasi data yang digunakan dalam penelitian tersebut adalah IEEE 802.11p. Salah satu skenario pada penelitian ini menggunakan keadaan lalu lintas pada daerah pedesaan. Dari hasil skenario tersebut didapat hasil maksimum *latency* tiap kendaraan bernilai 0.018 detik, lebih kecil dari 0.5 detik dari waktu yang seharusnya diperlukan untuk mencegah terjadinya kecelakaan berikutnya. Sedangkan jarak maksimum data yang dikirimkan adalah 1056.83m lebih kecil dari 1000 meter yang dibutuhkan (Phule, H., 2012).

Penelitian lainnya yaitu yang dilakukan oleh Scott J. Weiner pada penelitiannya yang berjudul "Feasibility of a 802.11 VANET Based Car Accident Alert System". Pada penelitian tersebut, penulis menganalisis tentang kemungkinan sebuah sistem yang dapat berjalan pada VANET dengan mengimplementasikan sebuah prototipe menggunakan standar IEEE 802.11g. Sistem perangkat keras yang dibangun terdiri atas GPS, *accelerometer* untuk mendeteksi bila terjadi tabrakan antar kendaraan, modul *wireless*, *speaker* dan *controller*. Sedangkan untuk perangkat lunak menggunakan uClinux. Dari hasil penelitian yang dilakukan memberikan hasil bahwa implementasi dari sistem dapat diterapkan pada kondisi yang hanya terdapat sebuah hop saja. Namun pada keadaan di jalan yang sebenarnya sistem tersebut dirasa tidak memungkinkan untuk diterapkan (Weiner, S. J., 2010).

2.2 Vehicular Ad-Hoc Network (VANET)

Vehicular ad-hoc network (VANET) merupakan jaringan *ad-hoc* sebagai bagian dari *Mobile Ad-hoc Network (MANET)*. Namun, yang membedakan adalah

node yang digunakan pada VANET adalah kendaraan yang tentunya berbeda dengan MANET yang memiliki node seperti *smartphone* (Ekka, 2013). Secara umum terdapat 3 tipe komunikasi pada VANET, pertama adalah komunikasi antara kendaraan (V2V), kedua komunikasi antara kendaraan dan infrastruktur (V2I), ketiga komunikasi infrastruktur dengan infrastruktur (I2I).

Standar komunikasi dalam VANET adalah *dedicated short range communication* (DSRC) dan IEEE 1609. DSRC digunakan untuk komunikasi jarak dekat pada V2V dan V2I yang kelebihanannya adalah *delay* komunikasi yang rendah dan data transfer yang tinggi. Standar dari DSRC berbeda-beda di Jepang, Eropa dan USA. Federal Communications Commission (FCC) di United States telah mengalokasikan 75 MHz dari spectrum berlisensi mulai 5.850 GHz hingga 5.925 GHz sebagai frekuensi *dedicated short range communication* untuk komunikasi dalam keamanan berkendara dan layanan informasi. Di Eropa, 2012 - 2020 MHz dari frekuensi yang tidak berlisensi di gunakan di Fleetnet dan dari berita terbaru, European Telecommunications Standards Institute (ETSI) juga telah mengalokasikan 30 MHz spektrum radio pada 5.9 GHz untuk Intelligent Transport System (Kumar, R. & Dave, M., 2012).

VANET dapat di implementasikan untuk menyelesaikan beberapa masalah pada kondisi lalu lintas saat ini. Beberapa penerapan dari VANET adalah sebagai berikut:

- **Keamanan dalam berkendara**

Penerapan untuk keselamatan dalam lalu lintas, Tujuan dari aplikasi ini adalah untuk mengurangi kecelakaan lalu lintas dan membantu pengendara untuk meningkatkan keselamatan mereka. Aplikasi ini memberikan informasi kepada pengendara tentang adanya kecelakaan yang terjadi di sekitar pengendara.

- **Pengaturan Lalu Lintas**

Tujuan dari aplikasi ini adalah meningkatkan arus lalu lintas dan kordinasi lalu lintas dengan memberikan informasi lalu lintas terkini.

2.3 OMNeT++

OMNeT++ adalah sebuah *framework* simulasi jaringan *discrete-event* yang bertipe *object-oriented* yang dibuat oleh András Varga dari Technical University of Budapest, Department of Telecommunications (BME-HIT). OMNeT++ memiliki sebuah arsitektur yang umum, sehingga OMNeT++ dapat digunakan dalam banyak bidang permasalahan jaringan, antara lain:

- Pemodelan dari jaringan komunikasi dengan kabel dan nirkabel.
- Pemodelan protokol.
- Pemodelan dari queueing networks.
- Pemodelan dari multiprocessors dan sistem hardware terdistribusi lainnya.
- Memvalidasi arsitektur dari perangkat keras.

- Mengevaluasi performa dari perangkat lunak.

Bahasa pemrograman simulasi untuk OMNeT++ memanfaatkan bahasa NED, yaitu bahasa pemrograman yang digunakan untuk mendeskripsikan topologi jaringan. Dengan menggunakan NED, deskripsi jaringan dapat berisi deskripsi komponen penyusun jaringan yang bersifat modular. Untuk pengembangan model dilakukan dengan objek dan bahasa pemrograman C++. Pada OMNeT++ dimungkinkan pula penggunaan Java, namun untuk ini harus menggunakan komponen tambahan. Pemrograman antarmuka OMNeT++ cukup mudah karena menggunakan Eclipse. Basis antar muka grafis juga tersedia untuk memprogram NED. Pengguna OMNeT++ dapat memilih pemrograman secara tekstual atau grafis (dapat berpindah antara basis teks dan grafis). OMNeT++ juga menyediakan contoh algoritma tertentu untuk pengiriman data antar node (Karyono, Martoyo, I. & Uranus, H. P., 2008).

OMNet++ juga menyediakan infrastruktur dan *tools* untuk memprogram simulasi sendiri. Pemrograman OMNet++ bersifat *object-oriented* dan bersifat hierarki. Objek-objek yang besar dibuat dengan cara menyusun objek-objek yang lebih kecil. Objek yang paling kecil disebut *simple module*, akan memutuskan algoritma yang akan digunakan dalam simulasi tersebut.

Berbagai tipe objek pada OMNet++ adalah :

1. **Module (Simple Module dan Compound Module)** adalah objek yang kita buat, kita program dan kita susun. *Compound Module* adalah sebuah modul yang dibuat dengan cara menggabungkan beberapa *Simple Module*.
2. **Gate** adalah pintu keluar / masuk *message*. Setiap modul hanya bisa berinteraksi dengan modul lainnya melalui *gate*.
3. **Message** adalah komunikasi yang dilakukan antar modul. *Message* adalah konsep inti dari simulasi OMNet++. Sebuah modul bisa mengirimkan *message* pada modul lain atau dirinya sendiri (*self message*).
4. **Connection** adalah jalur tempat dimana *message* mengalir. Disini kita bisa mendefinisikan parameter / variabel yang berkaitan dengan koneksi, misalnya hambatan udara, *datarate* dan lain sebagainya (Andras Varga and OpenSim Ltd, 2014).

2.4 Simulation of Urban Mobility (SUMO)

Simulation of Urban Mobility (SUMO) merupakan sebuah simulator untuk keadaan lalu lintas yang di kembangkan oleh Institute of Transportation Systems at the German Aerospace Center yang dikembangkan menggunakan bahasa C++. Aplikasi ini merupakan *open source* dan berlisensi dibawah GPL.

SUMO memodelkan keadaan lalu lintas yang ada seperti jalur kendaraan, transportasi umum dan pejalan kaki. Pada SUMO terdapat beberapa fitur yang dapat menangani tugas seperti pencarian rute, visualisasi dan penghitungan emisi kendaraan.

- SUMO memiliki beberapa fitur yang diberikan, antara lain:
- *Microscopic simulation* - kendaraan, pejalan kaki dan transportasi umum di modelkan secara eksplisit.
- *Online interaction* – simulasi dapat dikendalikan melalui TRaCI.
- Simulation of multimodal traffic, sebagai contoh kendaraan, pejalan kaki dan transportasi umum.
- Waktu dari lampu lalu-lintas dapat diatur manual ataupun secara otomatis oleh SUMO.
- Tidak ada batasan akan besar sistem lalu lintas yang di buat ataupun jumlah kendaraan yang disumulasikan.
- Map dapat dibuat dengan mengimpor dari berbagai format seperti OpenStreetMap, VISUM, VISSIM, NavTeq.
- SUMO diimplementasikan dengan menggunakan C++ dan menggunakan *library* yang portabel (Institute of Transportation Systems, n.d).

2.5 Standar IEEE 802.11b

Standar 802.11b muncul di pasaran awal tahun 2000. IEEE 802.11b merupakan protokol pengembangan dari standar IEEE 802.11 yang asli, yang bertujuan untuk meningkatkan kecepatan hingga 5.5 Mb/s atau 11 Mb/s tapi tetap menggunakan frekuensi 2.4 GHz dan jangkauan hingga 300 meter di lingkungan terbuka. Pada implementasinya, kecepatan maksimum yang dapat diraih oleh standar IEEE 802.11b mencapai 5.9 Mb/s pada protokol TCP dan 7.1 Mb/s pada protokol UDP. Metode transmisi yang digunakannya adalah DSSS. Standar IEEE 802.11b menggunakan rentang frekuensi 2,4 GHz dengan 3 saluran radio yang tersedia. Standar protokol 802.11b yang masih digunakan hingga saat ini hanya memiliki kemampuan transmisi standar dengan 11 Mbps atau rata rata 5 Mb/s yang dirasa kinerjanya cukup lambat (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012).

Channel	Frekuensi Nominal (MHz)	Frekuensi Minimum (MHz)	Frekuensi Maksimum (MHz)
1	2412	2401	2423
2	2417	2405	2428
3	2422	2411	2433
4	2427	2416	2438
5	2432	2421	2443
6	2437	2426	2448
7	2442	2431	2453
8	2447	2436	2458
9	2452	2441	2463
10	2457	2446	2468
11	2462	2451	2473

Gambar 2.1 Channel Frekuensi 2.4GHz

Sumber (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012)

Kelebihan 802.11b diantaranya :

- Biaya relatif murah.
- Standar 802.11b saat ini yang paling banyak digunakan.
- Menawarkan throughput maksimum dari 11 Mbps (6 Mbps dalam praktek) dan jangkauan hingga 300 meter di lingkungan terbuka dengan menggunakan rentang frekuensi 2,4 GHz, dengan 3 saluran radio yang tersedia.
- Biaya produksi rendah, *range* sinyal bagus dan tidak mudah terhalang.

Kelemahan 802.11b diantaranya :

- Kecepatan maksimum transfer paling rendah (11 Mbps), mudah terinterferensi oleh sinyal dari peralatan lain yang menggunakan frekuensi 2.4 Ghz (Radino, N., 2013).

2.5.1 Frekuensi IEEE 802.11b

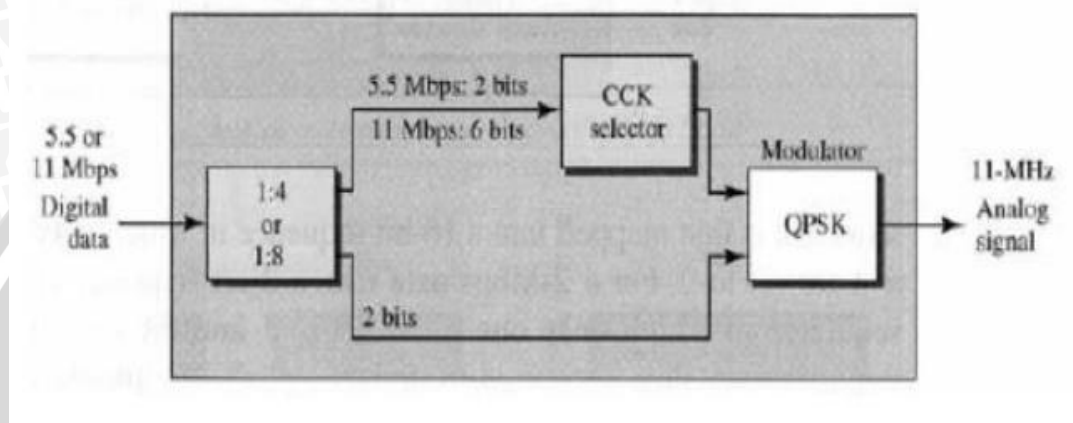
Frekuensi 2.4Ghz merupakan frekuensi yang digunakan oleh standar IEEE 802.11, 802.11b, 802.11g, dan 802.11n. Frekuensi 2.4Ghz yang dapat digunakan oleh WLAN dibagi-bagi menjadi *channel* yang berkisar dari 2.4000 sampai 2.4835 Ghz. Di negara US, *channel* yang dimiliki sebanyak 11, dan setiap *channel* tersebut mempunyai lebar pita 22 Mhz. Beberapa *channel overlap* / tumpang tindih dengan yang lainnya, hal ini menyebabkan masalah interferensi. Karena alasan inilah channel 1, 6, dan 11 adalah *channel* yang sering digunakan karena sinyalnya tidak overlap. Pada frekuensi 2.4Ghz modulasi yang digunakan adalah modulasi Direct Sequence Spread Spectrum (DSSS). Kecepatan transmisi datanya adalah 1 Mbps, 2 Mbps, 5.5 Mbps, dan 11 Mbps. Sedangkan Frekuensi 5 Ghz digunakan oleh standar 802.11a dan standar 802.11n yang merupakan protokol baru. Pada standar 802.11a, kecepatan transmisi data berkisar antara 6 Mbps sampai 54 Mbps. *Device* dengan protokol 802.11a tidak banyak digunakan, oleh karena itu penetrasi pasar untuk *device* standar 802.11a juga tidak sebanyak *device* standar 802.11b. Frekuensi 5Ghz juga dibagi-bagi menjadi beberapa *channel* dimana setiap channelnya selebar 20 Mhz. Total *channel* yang *non-overlap* adalah 23 *channel* pada frekuensi 5Ghz (Gunawan, E. K., Sugita, F. M., Abdurohim, M. W., 2012).

2.5.2 Modulasi IEEE 802.11b

IEEE 802.11b DSSS menggunakan metode high rate direct sequence spread spectrum (HR-DSSS) pada *band* frekuensi ISM 2.4 GHz. Prinsip kerja HR-DSSS mirip dengan DSSS kecuali pada metode *encodenya*, yang dinamakan dengan *complementary code keying (CCK)* (Gunawan, E. K., Sugita, F. M., Abdurohim, M. W., 2012).

Complementary Code Keying (CCK) merupakan skema modulasi untuk *datarate* 5.5 dan 11Mbps pada frekuensi 2.4GHz yang dimana modulasi CCK dispesifikasikan oleh standar tingkat tinggi dari 802.11 (Pearson, B., 2000). CCK melakukan *encode* 4 atau 8 bit menjadi satu simbol CCK supaya kompatibel dengan DSSS. HR-DSSS mendefinisikan empat kecepatan data yaitu 1, 2, 5.5, dan

11 Mbps. Dua kecepatan data yang pertama teknik modulasinya sama dengan DSSS konvensional (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012). DSSS konvensional menggunakan saluran yang lebih konvensional. Tiap saluran merupakan suatu band frekuensi yang bersebelahan yang lebarnya 22 MHz. Saluran 1, misalnya, bekerja dari frekuensi 2,401 GHz sampai 2,432 GHz ($2,412 \text{ GHz} \pm 11 \text{ MHz}$); saluran 2 bekerja dari 2,406 sampai 2,429 GHz ($2,417 \pm 11 \text{ MHz}$), dan seterusnya (Firmansyah, A., 2008). Versi 5.5 Mbps menggunakan BPSK dan mengirimnya pada 1.375 Mbaud/s dengan *encoding* 4-bit CCK. Versi 11 Mbps memakai QPSK dan mengirimnya di 1.375 Mbaud/s (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012).



Gambar 2.2 Skema Lapisan Fisik 802.11b

Sumber (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012)

Untuk mengirim data dengan teknik DSSS, digunakan teknik *Chipping Sequence* untuk mengatasi kemungkinan interferensi. *Chipping Sequence* merupakan proses mengirim 1 bit data dengan sederetan data binary. Misalnya data "1001" maka akan dikirim 4 Chipping sequence yaitu : "1001" => "00110011011 11001100100 11001100100 00110011011".

Karena data *chipping* yang dihasilkan lebih banyak daripada data sebenarnya, maka kecepatan *chipping* lebih besar daripada kecepatan data. Ada beberapa metode *chipping* yang digunakan yaitu metode *Barker Code* dan *Complementary Code Keying*. Untuk Mencapai kecepatan transmisi 1 Mbps dan 2 Mbps, 802.11 menggunakan *Barker Code* yang mendefinisikan penggunaan 11 *chips* saat *encoding* data. Sebanyak 11 *Chip Barker Code* yang digunakan dalam 802.11 adalah 10110111000. *Barker Code* merupakan metode yang ideal untuk memodulasi gelombang radio, sedangkan untuk mencapai kecepatan transmisi 5.5 Mbps dan 11 Mbps, DSSS menggunakan metode lain yang memungkinkan standar 802.11 untuk mencapai kecepatan 5.5 Mbps hingga 11 Mbps. *Complementary Code Keying (CCK)* digunakan untuk mencapai 5.5 Mbps dan 11 Mbps. CCK menggunakan beberapa seri kode yang di sebut Complementary Sequences yang terdiri dari 64 kode unik (Gunawan, E. K., Sugita, F. M., Abdurrohman, M. W., 2012).

2.6 INET Framework

INET Framework merupakan model *library open-source* yang digunakan untuk lingkungan simulasi pada OMNeT++. INET mengembangkan protokol dan model lainnya yang berguna untuk penelitian dan pengembangan tentang jaringan komunikasi. INET sangat berguna pada saat merancang dan melakukan validasi protokol baru maupun mengembangkan skenario yang masih baru ataupun sudah ada sebelumnya (INET Framework).

INET terdiri dari beberapa model untuk *stack* internet (TCP, UDP, IPv4, IPv6, OSPF, BGP, dll), protokol *link layer wired* dan *wireless* (Ethernet, PPP, IEEE 802.11, dll), protokol MANET, DiffServ, MPLS dengan LDP dan sinyal RSVP-TE, dan masih banyak komponen dan protokol lainnya. Pada INET, *routing* statis dapat diatur menggunakan *network autoconfigurators*, atau juga dapat menggunakan implementasi protokol routing yang telah tersedia (INET Framework).

Beberapa *framework* simulasi yang lainnya juga menggunakan INET sebagai dasar pengerjaannya yang kemudian dilakukan pengembangan menuju arah yang lebih spesifik, seperti *vehicular network*, jaringan *peer-to-peer*, atau LTE (INET Framework).

Infrastruktur INET sangat berguna untuk pengembangan OMNeT++. Memanfaatkan layanan yang disediakan pada *library* dan *kernel* simulasi OMNeT++ akan menghasilkan parameter kinerja dan hasil pengujian yang kemudian dapat dianalisis dan dikembangkan.

Beberapa fitur INET Framework:

- Implementasi *OSI Layer* (*physical, link-layer, network, transport, application*)
- Jaringan IPv4/IPv6 (atau mengembangkan sendiri *network layer*)
- Protokol *transport layer* : TCP, UDP, SCTP
- Routing protokol (*ad-hoc* dan *wired*)
- *Wired / Wireless Interfaces* (Ethernet, PPP, IEEE 802.11, dll)
- *Physical layer* dengan level *scalable* yang detail
- Support berbagai pemodelan lingkungan fisik (hambatan untuk propagasi radio, dll) (INET Framework)

2.7 End-to-End Delay

End-to-End Delay adalah waktu yang diperlukan oleh suatu paket data yang berasal dari *source node* hingga mencapai *destination node*. *End-to-End Delay* secara tidak langsung berhubungan dengan kecepatan transfer data suatu jaringan. Beberapa kontribusi jaringan dalam *End-to-End Delay* adalah *delay* transmisi, *delay* propagasi dan *delay* queue (Muslim, M. A., 2007).

Delay transmisi adalah waktu yang diperlukan oleh sebuah paket data untuk melintasi suatu media. *Delay* transmisi dipengaruhi oleh kecepatan media dan besar paket data yang dirumuskan sebagai berikut (Muslim, M. A., 2007) :

$$DT = \frac{N}{R}$$

Keterangan:

DT = *Delay Transmisi*

N = Besar Data yang Dikirim (Bytes)

R = *Bitrate* (bps)

Delay Propagasi adalah waktu yang diperlukan oleh suatu info untuk melintasi suatu garis dengan jarak tertentu. *Delay propagasi* ditentukan oleh jarak dan kecepatan media transmisi yang dapat dirumuskan sebagai berikut (Muslim, M. A., 2007):

$$DP = \frac{d}{s}$$

Keterangan:

DP = *Delay Propagasi*

d = Jarak

s = Kecepatan Media Transmisi

Delay Queue merupakan waktu yang diperlukan sebuah node atau perangkat jaringan untuk berada di dalam sebuah antrian hingga dapat melakukan suatu pemrosesan atau pengiriman paket (Muslim, M. A., 2007).

2.8 Throughput

Throughput adalah *bandwidth* aktual yang terukur pada suatu ukuran waktu tertentu dalam suatu hari menggunakan rute internet yang spesifik ketika sedang melakukan unduhan suatu file. *Throughput* dapat diukur dalam satuan bit per detik (bps) atau paket per detik (pps). *Throughput* juga mengukur efisiensi dan efektivitas kinerja protokol dan menentukan kinerja jaringan dari satu node ke tujuan (Dewo, E. S., 2003).

Bandwidth dan *throughput* memiliki cara pengukuran yang berbeda. *Bandwidth* adalah jumlah bit yang dapat dikirimkan dalam satu detik. Berikut adalah rumus dari *bandwidth* (Dewo, E. S., 2003):

$$bandwidth = \frac{\Sigma bits}{s}$$

Keterangan:

$\Sigma bits$ = jumlah bits

s = detik

Sedangkan *throughput* walau pun memiliki satuan dan rumus yang sama dengan *bandwidth*, tetapi *throughput* lebih pada menggambarkan *bandwidth* yang sebenarnya (aktual) pada suatu waktu tertentu dan pada kondisi dan jaringan internet tertentu yang digunakan untuk mendownload suatu file dengan

ukuran tertentu. Berikut adalah formula perbandingan *throughput* dengan *bandwidth* (Dewo, E. S., 2003):

$$\text{waktu_download_terbaik} = \frac{\text{ukuran_file}}{\text{bandwidth}}$$

$$\text{waktu_download_typical} = \frac{\text{ukuran_file}}{\text{throughput}}$$

2.9 Packet Loss

Packet Loss merupakan sebuah nilai yang menggambarkan kondisi jumlah total paket yang hilang atau tidak sampai ketujuan. Salah satu penyebab *packet loss* adalah antrian yang melebihi kapasitas *buffer* pada setiap *node* (Rizaldi et al., 2010).

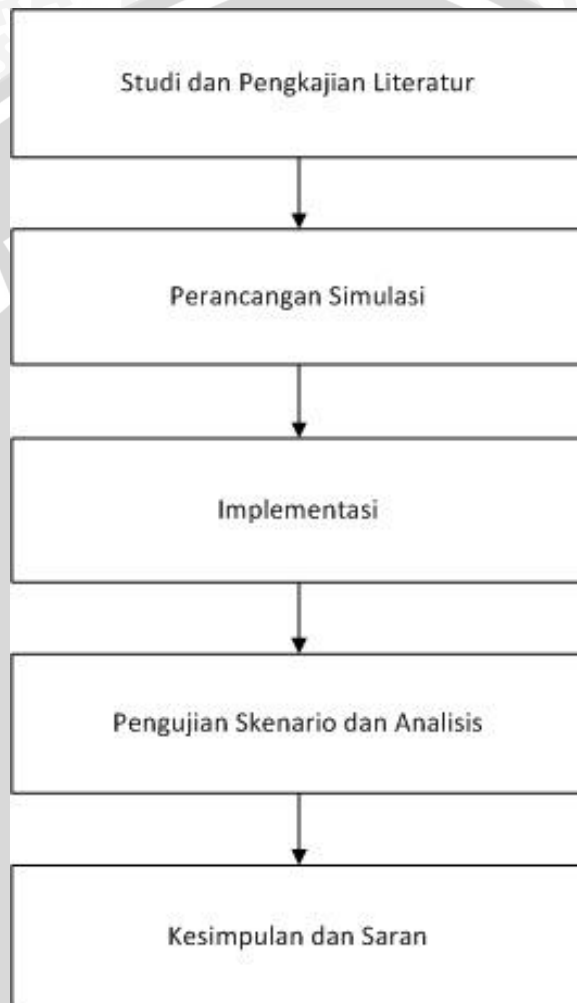
Beberapa penyebab terjadinya *packet loss* yaitu:

- *Congestion*, disebabkan terjadinya antrian yang berlebih dalam suatu komunikasi jaringan.
- *Node* yang bekerja melebihi kapasitas *buffer*.
- *Memory* yang terbatas pada *node*.
- *Policing* atau control terhadap jaringan untuk memastikan bahwa jumlah trafik yang mengalir sesuai dengan besarnya *bandwidth*. Jika besarnya trafik yang mengalir pada jaringan melebihi dari kapasitas *bandwidth* yang ada, maka *policing control* membuang kelebihan trafik yang ada (Rizaldi et al., 2010).

BAB 3 METODOLOGI

3.1 Metodologi Penelitian

Bab ini menjelaskan langkah-langkah yang akan dikerjakan dalam penyusunan skripsi, meliputi studi dan pengkajian literatur, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian dan analisis, kesimpulan dan saran.



Gambar 3.1 Flowchart Metode Penelitian

3.1.1 Studi Literatur

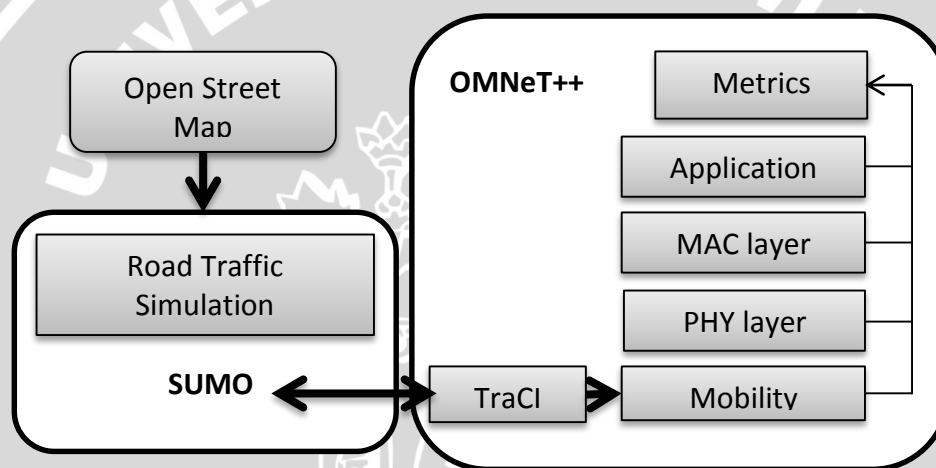
Studi literatur menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori – teori pendukung tersebut meliputi :

1. Kajian Pustaka
2. *Vehicular Ad-Hoc Network*
3. OMNeT++
4. *Simulation of Urban Mobility*
5. Standar IEEE 802.11b

- 5.1 Frekuensi IEEE 802.11b
- 5.2 Modulasi IEEE 802.11b
6. INET Framework
7. *End-to-End Delay*
8. *Throughput*
9. *Packet Loss*

3.1.2 Perancangan Simulasi

Perancangan simulasi dilakukan setelah semua kebutuhan simulasi didapatkan melalui tahap studi dan pengkajian literatur yang meliputi tentang teori dan kebutuhan perangkat lunak yang akan digunakan. Adapun kebutuhan perancangan simulasi pada penelitian ini ditunjukkan oleh gambar 3.2.



Gambar 3.2 Alur Kerja Simulator

Gambar 3.2 merupakan alur kerja pada simulator. Pada sistem terdapat tiga jenis perangkat lunak, yaitu SUMO versi 0.19.0, TraCI INET versi 2.5.0 dan OMNeT++ versi 4.6. SUMO merupakan simulator untuk mengatur keadaan lalu lintas, OMNeT++ bekerja sebagai simulator dalam komunikasi data sistem, sedangkan TraCI INET merupakan penghubung antara SUMO dan OMNeT++ agar dapat melakukan komunikasi satu sama lain.

Pada perancangan sistem diperlukan beberapa pemahaman penting sehingga hasil simulasi dapat tercapai. Sistem bekerja menggunakan teknologi VANET dengan menggunakan protokol 802.11b. Perancangan lalu lintas akan difokuskan pada kondisi perkotaan dengan terdapat beberapa kendaraan yang nantinya akan saling melakukan komunikasi satu sama lain.

Pada penelitian ini, peneliti mencoba beberapa skenario. Perbedaan skenario didasari dari jumlah node yang ada pada skenario tersebut. Parameter

yang akan menjadi bahan analisis pada penelitian ini sebanyak 3 parameter yang meliputi rata-rata *throughput*, *packet loss* dan *end-to-end delay*.

3.1.3 Implementasi

Implementasi merupakan penjelasan secara umum dari perancangan simulasi dan perancangan kebutuhan hingga hasil akhir analisis dari data yang diperoleh pada hasil simulasi.

Implementasi dilakukan dengan mengacu pada perancangan simulasi. Implementasi meliputi:

1. Perancangan peta dan simulasi keadaan lalu lintas pada SUMO

Perancangan peta yang digunakan pada simulasi dilakukan dengan mengimport dari Open Street Map yang kemudian diedit menggunakan aplikasi Java OpenStreetMap (JOSM). Sementara perancangan simulasi keadaan lalu lintas menggunakan *road traffic simulator Simulation of Urban Mobility* (SUMO) versi 0.19.0. Simulasi keadaan lalu lintas pada SUMO diatur sesuai dengan skenario yang diinginkan berdasarkan kondisi jalan raya (*highway*), luar kota (*rural*) dan perkotaan (*urban*).

2. Perancangan sistem jaringan kendaraan pada VANET

Simulasi jaringan VANET menggunakan TraCI INET versi 2.5.0 yang didukung oleh 2 *simulator* OMNET++ dan SUMO. OMNET++ adalah sebuah *event-based network simulator*. OMNET++ didesain untuk komunikasi *wireless* yang menggunakan pendekatan arsitektur berbasis komponen dan menggunakan bahasa pemrograman C++ serta Tcl/Tk untuk implementasi model dan simulasi (Andras Varga and OpenSim Ltd, 2014).

3. Memulai simulasi sesuai dengan skenario

Setelah perancangan peta, simulasi keadaan lalu lintas dan sistem jaringan kendaraan telah selesai dibuat dan dirancang maka implementasi program dapat dijalankan sesuai dengan skenario yang telah diatur sebelumnya.

3.1.4 Pengujian dan Analisis

Pengujian dan analisis dilakukan untuk mengetahui apakah kinerja dan performa keseluruhan simulasi yang telah dijalankan sesuai skenario yang ada. Pada penelitian ini terdapat beberapa skenario yang ada untuk menguji simulasi jaringan VANET dan menganalisis hasil yang diperoleh dari simulasi dengan parameter yang meliputi rata-rata *throughput*, *packet loss* dan *end-to-end delay*. Pada tahap pengujian terdiri dari beberapa skenario.

Tabel 3.1 List Skenario Pengujian

Skenario	Jumlah Percobaan	Jumlah Kendaraan	Besar Data	Maksimum Kecepatan Kendaraan
Kepadatan Kendaraan	5x	10, 20, 30, 40, 50	40 KB	Random
Besar Paket Data	6x	20	10 KB, 20 KB, 30 KB, 40 KB, 50 KB, 60 KB	20 km/jam
Kecepatan Kendaraan	6x	20	50 KB	20 km/jam, 40 km/jam, 60 km/jam, 80 km/jam, 100 km/jam, 120 km/jam

3.1.4.1 Pengujian terhadap kepadatan kendaraan

Pada **Tabel 3.1** skenario pengujian kepadatan kendaraan ini membandingkan pengaruh jumlah node atau kepadatan kendaraan terhadap parameter kinerja. Pada percobaan ini terdiri atas lima kali pengujian dengan besar data yang dikirim adalah 40 KB dan terdapat sejumlah variasi parameter kepadatan kendaran yang digunakan.

3.1.4.2 Pengujian terhadap besar paket data

Pada **Tabel 3.1** skenario pengujian besar paket data bertujuan untuk membandingkan pengaruh besar paket data yang diberikan pada tiap percobaan yang berbeda-beda terhadap parameter kinerja. Total kendaraan pada simulasi ini adalah 20 kendaraan.

3.1.4.3 Pengujian terhadap kecepatan kendaraan

Pada **Tabel 3.1** skenario kecepatan kendaraan bertujuan untuk membandingkan pengaruh kecepatan pada kendaraan terhadap parameter kinerja. Total kendaraan pada simulasi ini adalah 20 kendaraan dengan besar paket yang dikirimkan adalah 50 KB yang dikirimkan.

3.1.5 Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan implementasi, dan pengujian sistem yang dibangun telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memberikan pertimbangan atas pengembangan penelitian lebih lanjut.

BAB 4 HASIL

Bab ini membahas mengenai hasil dari tahapan perancangan dan implementasi pada simulator OMNeT++ terhadap analisis jaringan VANET yang disesuaikan dengan analisis kebutuhan serta perancangan simulasi. Implementasi dan pengujian mencakup penjelasan tentang spesifikasi sistem, konfigurasi simulator dan prosedur simulasi.

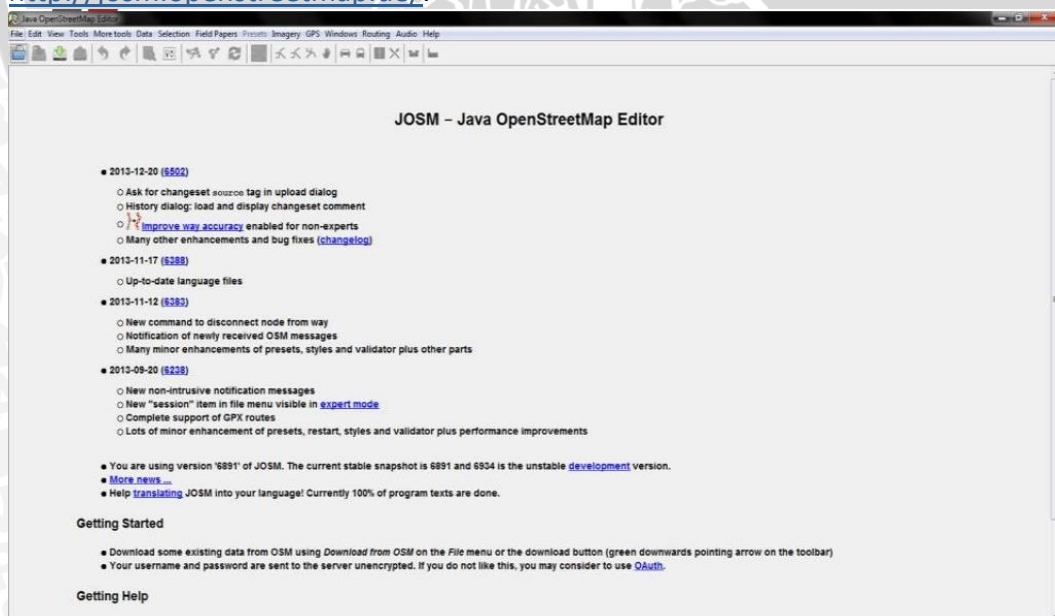
4.1 Rancangan Implementasi

Secara umum, sistem yang dibangun adalah simulasi dan analisis teknologi jaringan VANET dengan topologi yang dibangun sesuai dengan keadaan lalu lintas pada jalan Kalpataru, Kota Malang. Dengan membandingkan hasil simulasi dari parameter yang telah ditentukan. Proses perancangan ini berisikan kebutuhan yang dibutuhkan untuk membangun simulasi yang sesuai dengan penelitian ini.

4.1.1 Perancangan Peta dan Keadaan lalu Lintas pada SUMO

Perancangan peta yang digunakan pada simulasi dapat dilakukan dengan mengimport dari Open Street Map yang kemudian diedit menggunakan aplikasi Java OpenStreetMap (JOSM).

Java OpenStreetMap (JOSM) merupakan software pemetaan yang digunakan untuk melakukan editing (menambahkan, memperbaiki, dan memperbaharui) data spasial ke dalam OpenStreetMap (OSM) menggunakan bahasa Java 7 (Edgewall Software, 2013). Aplikasi JOSM khusus dikembangkan untuk OpenStreetMap (OSM) dan dapat diunduh secara gratis di alamat website <http://josm.openstreetmap.de/>.



Gambar 4.1 Tampilan Aplikasi Java OpenStreetMap (JOSM)

Gambar 4.1 merupakan tampilan awal aplikasi Java OpenStreetMap (JOSM) yang digunakan untuk mengedit map yang akan digunakan pada simulasi program.

4.1.1.1 Perancangan Peta Menggunakan Open Street Map

Perancangan peta yang digunakan pada simulasi menggunakan data peta di daerah Jalan Kalpartaru, Malang berdasarkan Open Street Map (openstreetmap.org).

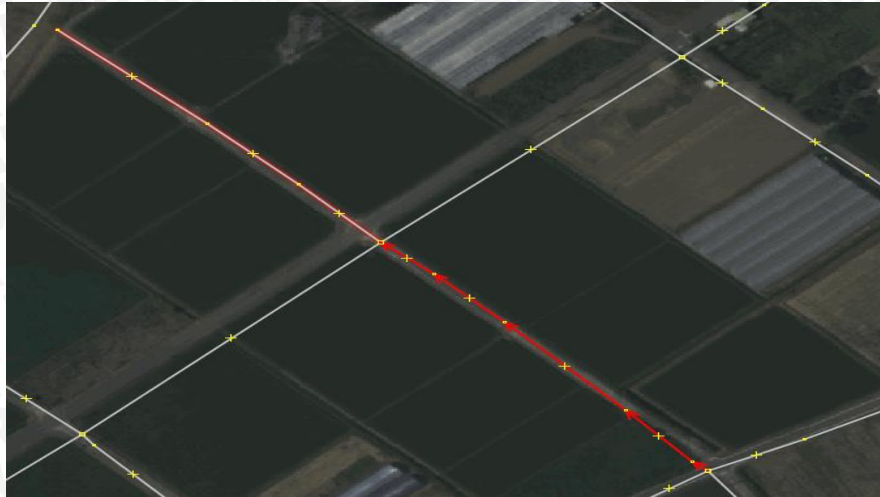


Gambar 4.2 Peta yang akan digunakan

Sumber (openstreetmap.org)

Gambar 4.2 merupakan peta yang akan digunakan pada simulasi berdasarkan Open Street Map. Pada simulasi kendaraan akan berjalan dari empat titik dan melewati perempatan Jalan Kalpartaru, Jalan Cengger Ayam dan Jalan Melati.

Setelah map berhasil diimport pada Open Street Map, dapat dilakukan *cropping* map pada aplikasi Java Open Street Map (JOSM) seperti pada gambar 4.3.



Gambar 4.3 Proses Cropping Map pada JOSM

Gambar 4.3 merupakan proses *cropping* map pada JOSM. Proses ini berfungsi untuk menyeleksi bagian mana saja dari map yang digunakan pada simulasi pengujian.

```
G:\map>netconvert --osm map.osm
Warning: The traffic light '2412873946' does not control any links; it will not
be build.
Warning: Could not build program '0' for traffic light '2412873946'
Success.
```

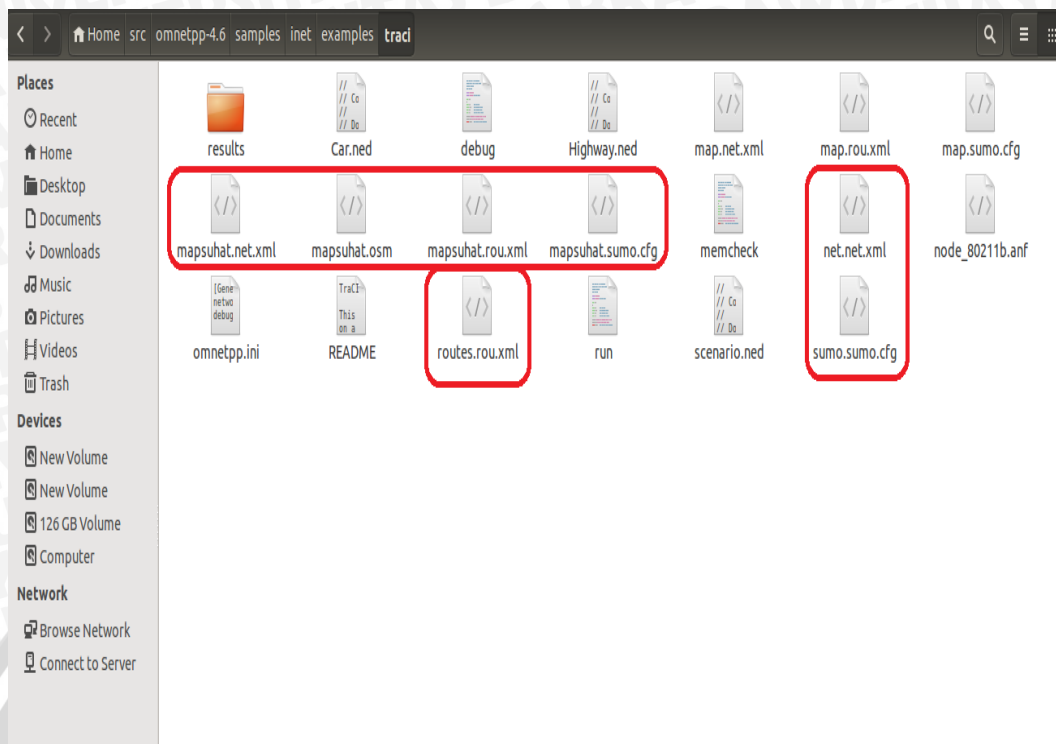
Gambar 4.4 Konversi file OSM

Gambar 4.4 merupakan proses konversi peta dari Open Street Map. Peta yang diambil dari Open Street Maps berbentuk file .osm, sehingga perlu dikonversi terlebih dahulu menjadi file .xml untuk dapat digunakan pada simulator SUMO.

4.1.1.2 Pengaturan TraCI pada INET Framework

TraCI INET merupakan *library* pada OMNET++ yang digunakan pada simulasi ini dan menjadi penghubung antara SUMO dan OMNeT++ agar dapat melakukan komunikasi satu sama lain.

Setelah melakukan tahapan konversi peta dari OpenStreetMap dengan file berekstensi dasar .osm menjadi file .xml, selanjutnya file tersebut dimasukkan ke dalam file *library* dari TraCI INET. (.net.xml, .rou.xml, .sumo.cfg dan .osm) yang kemudian dilakukan *build project* pada simulator OMNET++ yang berfungsi untuk membangun dan menjalankan simulasi jika telah siap digunakan.



Gambar 4.5 Gambar file berekstensi .net.xml, .rou.xml, .sumo.cfg dan .osm

Gambar 4.5 merupakan gambar file dari *library* TraCI INET yang merupakan file berekstensi .net.xml, .rou.xml, .sumo.cfg dan .osm yang difungsikan untuk menampilkan peta pada simulasi.

File ekstensi .net.xml berisikan kooordinat *edge id* sesuai dengan peta yang diambil dari OpenStreetMap (OSM). File ekstensi .osm berupa peta asli yang belum dikonversi menjadi file berekstensi .net.xml. File .rou.xml berisikan pengaturan rute yang diinginkan sesuai dengan kondisi jalan pada peta. File .sumo.cfg berisi pengaturan koneksi antara file ekstensi .net.xml, .osm dan .rou.xml.

Kemudian dilakukan pengaturan pada file Car.ned yang berada di dalam TraCI INET. Pada file ini dilakukan penambahan baris kode untuk menampilkan nilai *throughput* seperti yang ditunjukkan pada gambar 4.6.

```
import inet.util.ThruputMeter;
import inet.util.ThruputMeteringChannel;

    thruptMeter_tx: ThruputMeter {
        @display("p=280,229;is=s");
    }
    thruptMeter_rx: ThruputMeter {
        @display("p=415,252;is=s");
    }
connections allowunconnected:
    udp.appOut++ --> app.udp$i;
    udp.appIn++ <-- app.udp$o;

    udp.ipOut --> networkLayer.transportIn++;

    udp.ipIn <-- thruptMeter_rx.out;
    thruptMeter_rx.in <-- networkLayer.transportOut++;

    wlan.upperLayerOut --> networkLayer.ifIn[0];
    wlan.upperLayerIn <-- networkLayer.ifOut[0];

    radioIn --> wlan.radioIn;
}
```

Gambar 4.6 Konfigurasi untuk Menampilkan Nilai Throughput

Gambar 4.6 merupakan baris kode untuk menampilkan nilai *throughput* dari simulasi yang kemudian dapat dilihat pada hasil pengujian. Menambah baris kode *import inet.util.ThruputMeter* dan *import inet.util.ThruputMeteringChannel* bertujuan untuk menginisiasi parameter penghitungan *throughput*. Sedangkan, pada baris kode *udp.ipIn <-- thruptMeter_rx.out* dan *thruputMeter_rx.in <-- networkLayer.transportOut++* berfungsi untuk merekam setiap aliran data yang dikirimkan dengan UDP melalui *network layer*.

Selanjutnya melakukan pengaturan pada file Highway.ned di *library* TraCI INET yang dapat dilihat pada gambar 4.7.

```
import inet.examples.traci.Car;
import inet.networklayer.autorouting.ipv4.IPv4NetworkConfigurator;

configurator: IPv4NetworkConfigurator {
    parameters:
        config = xml("<config><interface hosts='*' address='145.236.x.x' netmask='255.255.0.0' /><");
        @display("p=50,100");
    }
}
```

Gambar 4.7 Import Traci Car dan IPv4NetworkConfigurator

Gambar 4.7 merupakan baris kode untuk mengimport Traci Car yang sebelumnya sudah ditambah baris kode tertentu (lihat pada gambar 4.5). Fungsi IPv4NetworkConfigurator adalah untuk menandai alamat IP yang digunakan dan

mengatur *routing* statis untuk jaringan IPv4. Pada gambar 4.6 alamat IP yang digunakan untuk pengiriman data yakni “145.236.x.x”.

Apabila langkah-langkah diatas telah selesai dilakukan, langkah selanjutnya adalah menginisialisasi pengaturan *End-to-End Delay* dan *throughput* agar nantinya apabila dilakukan *run* simulasi hasilnya dapat muncul pada laporan hasil pengujian. Pengaturan tersebut dapat dilihat pada gambar 4.8.

```
** .udpApp.endToEndDelay.scalar-recording = true  
** .udpApp.endToEndDelay.vector-recording = true  
*.thruputMeter_rx.vector-recording = true
```

Gambar 4.8 Konfigurasi Menampilkan dan Record Hasil Delay dan Throughput

Gambar 4.8 merupakan konfigurasi untuk menampilkan dan merekam hasil *End-to-End Delay* (berupa *scalar* dan *vector*) dan *throughput* berupa *vector* setelah *run* simulasi berhasil dilakukan. Data yang didapat secara *scalar* dikumpulkan secara manual melalui objek *histogram* ketika simulasi berhenti dengan sukses, bukan ketika simulasi berhenti ketika ada suatu kesalahan.

4.1.2 Perancangan Sistem Jaringan Pada Node Kendaraan

Pada sub bab ini akan dijelaskan mengenai perancangan node yang akan digunakan pada simulasi. Pada sistem ini akan hanya terdapat satu jenis node yaitu *mobile node* yang berupa kendaraan.

4.1.2.1 Konfigurasi Sistem Kendaraan

Pada OMNeT++ terdapat pengaturan jaringan pada kendaraan tersebut. Pada sistem ini node kendaraan berfungsi untuk menerima pesan yang dikirimkan kendaraan lain dan kemudian meneruskan pesan tersebut kepada node kendaraan lain yang ada disekitarnya.

```

# TraCIScenarioManager
*.manager.updateInterval = 1s
*.manager.host = "localhost"
*.manager.port = 8888
*.manager.moduleType = "inet.examples.traci.Car"
*.manager.moduleName = "host"
*.manager.moduleDisplayString = ""
*.manager.autoShutdown = true
*.manager.margin = 25

# nic settings
**.wlan.bitrate = 11Mbps
**.wlan.opMode = "b"
**.wlan.mac.wifiPreambleMode = "SHORT"
**.wlan.mgmt.frameCapacity = 999999999
**.wlan.mgmtType = "Ieee80211MgmtAdhoc"
**.wlan.mac.address = "auto"
**.wlan.mac.maxQueueSize = 99999999
**.wlan.mac.rtsThresholdBytes = 3000B
**.wlan.mac.retryLimit = 7
**.wlan.mac.cwMinData = 7
**.wlan.mac.cwMinBroadcast = 31
**.wlan.mgmt.authenticationTimeout = 100000000s

**.wlan.radio.transmitterPower = 2mW
**.wlan.radio.thermalNoise = -110dBm
**.wlan.radio.sensitivity = -85dBm
**.wlan.radio.pathLossAlpha = 2
**.wlan.radio.snirThreshold = 4dB

```

Gambar 4.9 Konfigurasi Sistem Jaringan Kendaraan

Gambar 4.9 merupakan konfigurasi dari sistem jaringan pada kendaraan. Pada kendaraan tidak melakukan pengiriman data secara periodik dikarenakan tergantung dari pengiriman dan penerimaan data dari kendaraan lainnya.

Pada gambar 4.9 ada beberapa parameter yang perlu diatur untuk keperluan simulasi pada sistem jaringan kendaraan. Pengaturan *wlan.bitrate* = 11Mbps digunakan untuk mengatur *bitrate* sebesar 11Mbps yang merupakan standar *bitrate* dari protokol 802.11b. *Wlan.opMode* = "b" maksudnya adalah protokol yang digunakan pada pengujian ini adalah b (802.11b). *Wlan.mgmt.frameCapacity* = 999999999 maksudnya adalah nilai 999999999 merupakan kapasitas frame yang digunakan pada simulasi. *Wlan.mac.maxQueueSize* = 99999999 maksudnya adalah nilai 99999999 merupakan besaran maksimum pada antrian data yang dikirimkan. Sedangkan pengaturan *wlan.mgmt.authenticationTimeout* = 100000000s maksudnya adalah waktu maksimum untuk proses autentikasi pengiriman data selama 100000000 detik.

4.1.3 Konfigurasi Standar IEEE 802.11b

Pada sub bab ini akan dijelaskan bagaimana cara untuk mengatur penggunaan 802.11b yang akan digunakan pada simulasi. Pada TraCI INET terdapat model dari IEEE 802.11b yang digunakan dalam VANET. Tabel 4.1 merupakan konfigurasi dari IEEE 802.11b yang digunakan pada simulasi. Data Rate adalah 11Mbps dan berada pada channel frekuensi 2.4 GHz.

Tabel 4.1 Parameter 802.11b

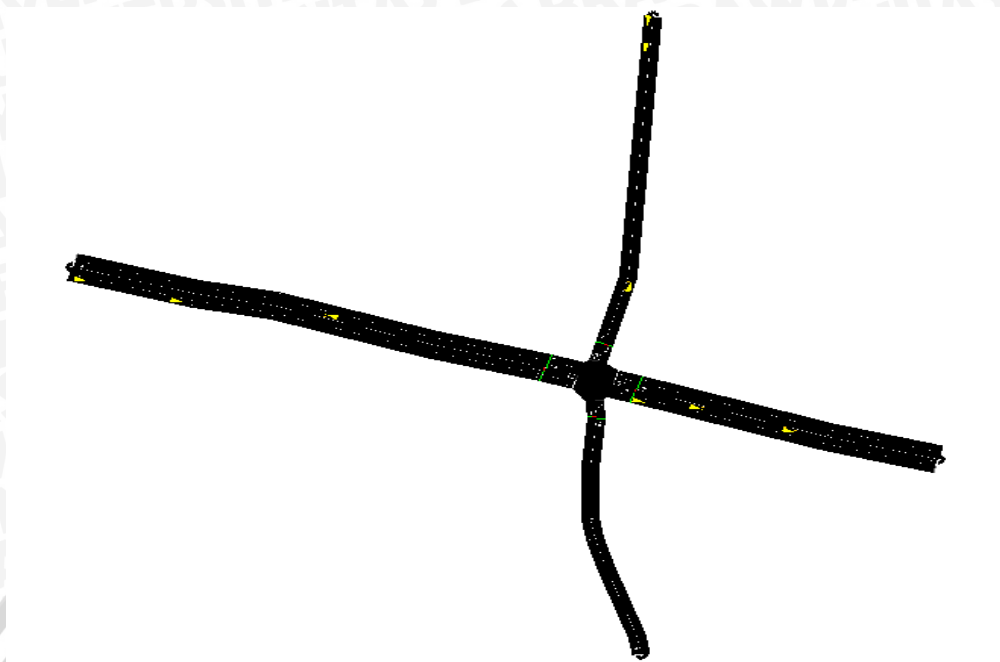
Parameter	Value
<i>Tx Power physical layer</i>	2mW
<i>Carrier frequency</i>	2.4 GHz
<i>Bitrate</i>	11Mbps
<i>Sensitivity</i>	-85dBm

4.2 Pengujian

Bab ini membahas proses pengujian terhadap skenario yang digunakan pada simulasi. Pada pengujian terdapat beberapa skenario, antara lain: Pengujian pengaruh kepadatan kendaraan, Pengujian pengaruh besar paket data dan Pengujian pengaruh kecepatan kendaraan terhadap pengaruh parameter kinerja.

4.2.1 Pengujian Terhadap Kepadatan

Pada sub bab ini bertujuan untuk menguji pengaruh kepadatan kendaraan terhadap parameter kinerja pada vanet. Map yang di gunakan dalam pengujian menggunakan map pada daerah perempatan jalan kalpataru.



Gambar 4.10 Peta untuk Pengujian

Gambar 4.10 merupakan peta yang digunakan dalam pengujian skenario ini. Peta yang digunakan pada pengujian ini berdasarkan OpenStreetMap. Pada skenario terdapat empat jenis kendaraan dengan jalur yang berbeda.

```
<flow id="flow0" type="CarA" route="route0" begin="0" period="3" number="5"/>
<flow id="flow1" type="CarB" route="route1" begin="0" period="3" number="5"/>
```

Gambar 4.11 Konfigurasi untuk 10 jumlah kendaraan

Gambar 4.11 merupakan konfigurasi kendaraan pada SUMO dengan jumlah kendaraan adalah 10 (sepuluh).

```
<flow id="flow0" type="CarA" route="route0" begin="0" period="3" number="10"/>
<flow id="flow1" type="CarB" route="route1" begin="0" period="3" number="10"/>
```

Gambar 4.12 Konfigurasi untuk 20 jumlah kendaraan

Gambar 4.12 merupakan konfigurasi kendaraan pada SUMO dengan jumlah kendaraan adalah 20 (dua puluh).

```
<flow id="flow0" type="CarA" route="route0" begin="0" period="3" number="15"/>
<flow id="flow1" type="CarB" route="route1" begin="0" period="3" number="15"/>
```

Gambar 4.13 Konfigurasi untuk 30 jumlah kendaraan

Gambar 4.13 merupakan konfigurasi kendaraan pada SUMO dengan jumlah kendaraan adalah 30 (tiga puluh).

```
<flow id="flow0" type="CarA" route="route0" begin="0" period="3" number="20"/>
<flow id="flow1" type="CarB" route="route1" begin="0" period="3" number="20"/>
```

Gambar 4.14 Konfigurasi untuk 40 jumlah kendaraan

Gambar 4.14 merupakan konfigurasi kendaraan pada SUMO dengan jumlah kendaraan adalah 40 (empat puluh).

```
<flow id="flow0" type="CarA" route="route0" begin="0" period="3" number="25"/>
<flow id="flow1" type="CarB" route="route1" begin="0" period="3" number="25"/>
```

Gambar 4.15 Konfigurasi untuk 50 jumlah kendaraan

Gambar 4.15 merupakan konfigurasi kendaraan pada SUMO dengan jumlah kendaraan adalah 50 (lima puluh).

4.2.2 Pengujian Terhadap Besar Data

Pada sub bab ini bertujuan untuk menguji pengaruh besar paket data yang di transmisi terhadap parameter kinerja pada VANET. Map yang di gunakan dalam pengujian menggunakan map jalan lurus dengan dua arah yang saling berlawanan.



Gambar 4.16 Map Jalan Lurus Dua Arah Berlawanan

Gambar 4.16 merupakan map yang digunakan dalam pengujian skenario besar data.

```
simple UDPBasicApp like IUDApp
{
    parameters:
        int localPort = default(-1); // local port (-1: use ephemeral port)
        string destAddresses = default(""); // list of IP addresses, separated by spaces ("": don't send)
        int destPort;
        volatile int messageLength @unit(B); // length of messages to generate, in bytes
        double startTime @unit(s) = default(this.sendInterval); // application start time (start of the first p
        double stopTime @unit(s) = default(-1s); // time of finishing sending, -1s means forever
        volatile double sendInterval @unit(s); // should usually be a random value, e.g. exponential(1)
        int timeToLive = default(-1); // if not -1, set the TTL (IPv4) or Hop Limit (IPv6) field of sent packet
        int typeOfService = default(-1); // if not -1, set the ToS (IPv4) or Traffic Class (IPv6) field of sent
        string multicastInterface = default(""); // if not empty, set the multicast output interface option on
        bool receiveBroadcast = default(false); // if true, makes the socket receive broadcast packets
        bool joinLocalMulticastGroups = default(false); // if true, makes the socket receive packets from all m
        @display("i=block/app");
        @signal[sendPk](type=cPacket);
        @signal[rcvdPk](type=cPacket);
        @statistic[rcvdPk](title="packets received"; source=rcvdPk; record=count,"sum(packetBytes)","vector(pac
        @statistic[sendPk](title="packets sent"; source=sendPk; record=count,"sum(packetBytes)","vector(packetB
        @statistic[rcvdPkLifetime](title="received packet lifetime"; source="messageAge(rcvdPk)"; unit=s; recor
    gates:
        input udpIn @labels(UDPControlInfo/up);
        output udpOut @labels(UDPControlInfo/down);
}
```

Gambar 4.17 Konfigurasi pengiriman pesan

Gambar 4.17 merupakan konfigurasi pengiriman data pada kendaraan. Perintah *int messageLength* digunakan untuk memanipulasi besar data dari paket yang akan dikirimkan dalam bentuk *byte*. Pada *source code* besar data diambil dari parameter *messageLength* yang bisa diatur pada *file omnetpp.ini* merupakan konfigurasi pada *omnetpp.ini*.

```

package inet.applications.traci;

//
// TraCI IVC Demo
//
simple TraCIDemo
{
    parameters:
        @display("i=block/app2");
        volatile int messageLength;

    gates:
        inout udp;
}

```

Gambar 4.18 File Ned untuk Kendaraan

Gambar 4.18 merupakan *source code* pada *file ned* untuk kendaraan. Untuk dapat membentuk variabel yang dapat dikonfigurasi pada *file omnetpp.ini* maka perlu dibentuk parameter pada *file ned*. *Int messageLength* pada gambar bertujuan untuk membuat variabel dengan nama "messageLength" yang dapat diatur pada *file omnetpp.ini*.

```

**.messageLength = 61440
**.udpApp.vector-recording = true
**.udpApp.scalar-recording = true
**.udpApp.endToEndDelay.scalar-recording = true
**.udpApp.endToEndDelay.vector-recording = true

```

Gambar 4.19 Konfigurasi pada omnetpp.ini

Gambar 4.19 merupakan konfigurasi pada *omnetpp.ini*. Pada konfigurasi terdapat perintah **.messageLength = 61440* yang bertujuan untuk memberikan nilai sebesar 61440 yang merupakan besar data yang akan dikirimkan oleh node kendaraan.

4.2.3 Pengujian Terhadap Kecepatan

Pada sub bab ini bertujuan untuk menguji pengaruh kecepatan kendaraan terhadap parameter kinerja pada *vanet*. Map yang di gunakan dalam pengujian menggunakan map berupa jalan lurus dengan dua arah yang saling berlawanan.

```

<vType accel="2.0" decel="5.0" id="CarA" length="2.0" maxSpeed="27.78" sigma="0.0" color="1,1,0"/>
<vType accel="2.0" decel="5.0" id="CarB" length="2.0" maxSpeed="27.78" sigma="0.0" color="1,0,0"/>

```

Gambar 4.20 Konfigurasi kecepatan pada SUMO

Gambar 4.20 merupakan konfigurasi pada SUMO yang digunakan untuk mengatur kecepatan pada kendaraan. Perintah *maxSpeed* adalah perintah untuk mengatur kecepatan maksimal kendaraan dalam m/s, sedangkan *accel* dan *decel* merupakan kemampuan akselerasi dan deselerasi pada kendaraan dalam m/s².

Pada SUMO setiap kendaraan akan melaju pada kecepatan maksimal yang diberikan kepada kendaraan kecuali terdapat halangan.



BAB 5 PEMBAHASAN

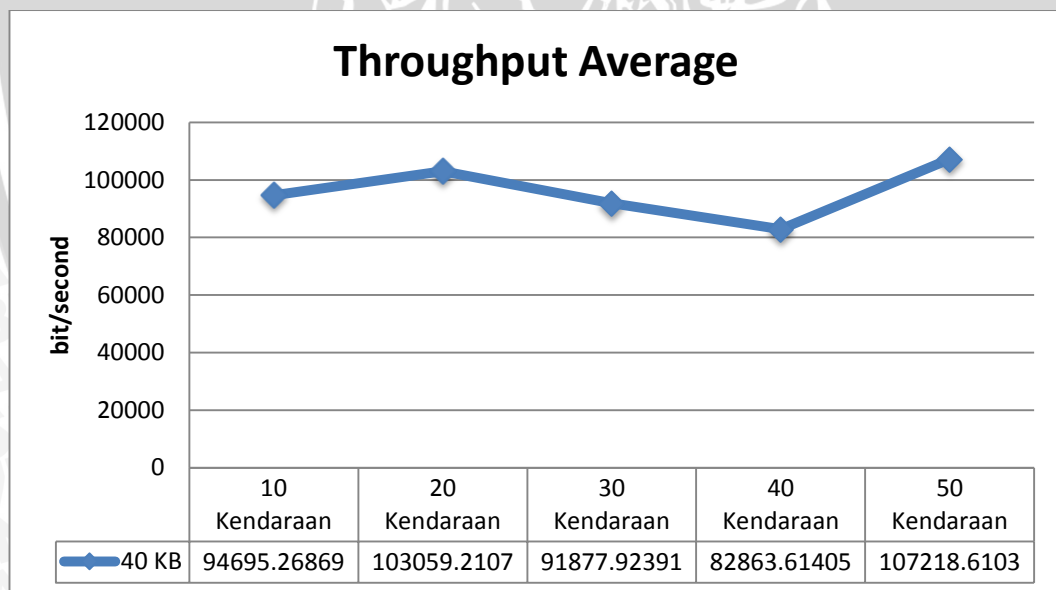
Pada bab ini akan membahas hasil dari pengujian dan analisis yang telah dirancang pada perancangan terhadap hasil simulasi dan analisis komunikasi pada kendaraan menggunakan protokol 802.11b. Pengujian dilakukan sesuai dengan skenario yang terdapat pada bab hasil. Hasil dari pengujian akan dianalisis sehingga dapat diketahui parameter apa saja yang mempengaruhi parameter kinerja.

5.1 Hasil

Hasil simulasi didapatkan dengan mengambil data pada pengujian pada simulator OMNeT++. semua hasil diperoleh dalam grafik dan data yang perlu dianalisis dengan deskripsi untuk dapat memberikan pemahaman tentang kinerja yang dihasilkan. Hasil yang diperoleh dari setiap skenario akan dianalisis pada sub bab ini.

5.1.1 Pengujian Terhadap Parameter Kepadatan Kendaraan

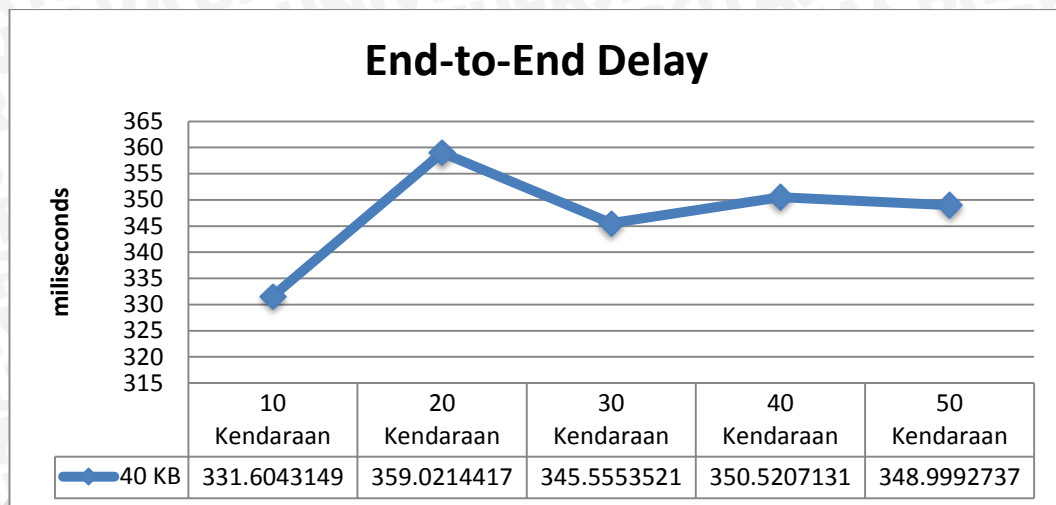
Pada sub bab ini terdapat hasil pengujian dari simulasi pengujian pengaruh kepadatan kendaraan pada parameter kinerja dalam vanet. Pada Pengujian terdapat tiga parameter yang diuji yaitu *Throughput*, *End-to-End Delay* dan *Packet loss*.



Gambar 5.1 Hasil Throughput pengujian kepadatan

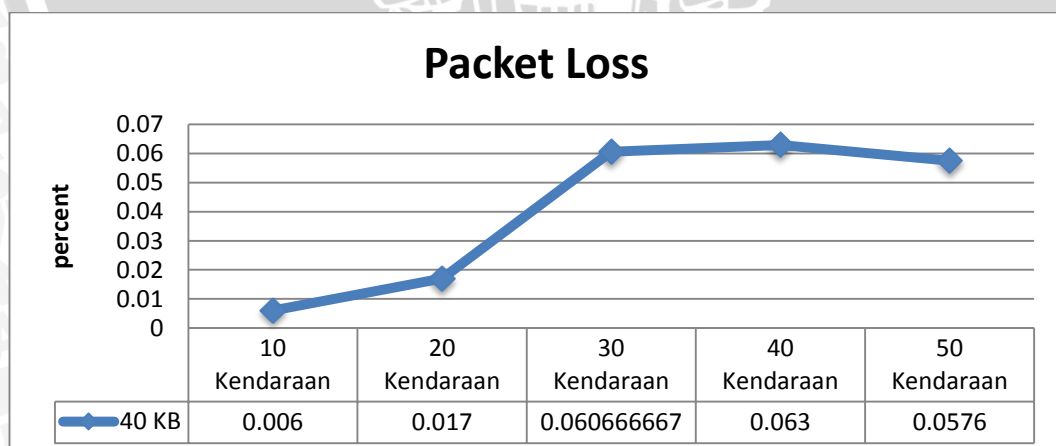
Gambar 5.1 merupakan hasil pengujian *throughput* terhadap kepadatan. Pada gambar dapat diketahui bahwa rata-rata *throughput* semakin naik jika jumlah kendaraan semakin banyak. Namun pada pengujian terjadi anomaly pada jumlah kendaraan 30 dan 40 menjadi lebih rendah daripada jumlah kendaraan 20 dikarenakan tipe *mobility* yang digunakan pada skenario pengujian ini adalah *random mobility*. Pengaruh jumlah kendaraan terhadap kinerja parameter terjadi

dikarenakan *channel* yang digunakan untuk pengiriman data harus terbagi pula untuk setiap kendaraan yang menerima data sehingga terjadi pengaruh terhadap nilai dari hasil *throughput*.



Gambar 5.2 End-to-End Delay pada pengujian kepadatan

Gambar 5.2 merupakan *End-to-End Delay* dari hasil pengujian terhadap kepadatan kendaraan. Berdasarkan gambar tersebut dapat diketahui bahwa nilai dari *End-to-End Delay* pada pengujian cenderung meningkat bila jumlah kendaraan meningkat. Namun pada pengujian terjadi anomaly pengujian pada jumlah kendaraan 30 menjadi lebih rendah daripada jumlah kendaraan 20 dan jumlah kendaraan 50 juga menjadi lebih rendah daripada jumlah kendaraan 40 dikarenakan tipe *mobility* yang digunakan pada skenario pengujian ini adalah *random mobility*. Pengaruh jumlah kendaraan terhadap kinerja parameter terjadi dikarenakan *channel* yang digunakan untuk pengiriman data harus terbagi pula untuk setiap kendaraan yang menerima data sehingga terjadi pengaruh terhadap nilai dari hasil *end-to-end delay*.



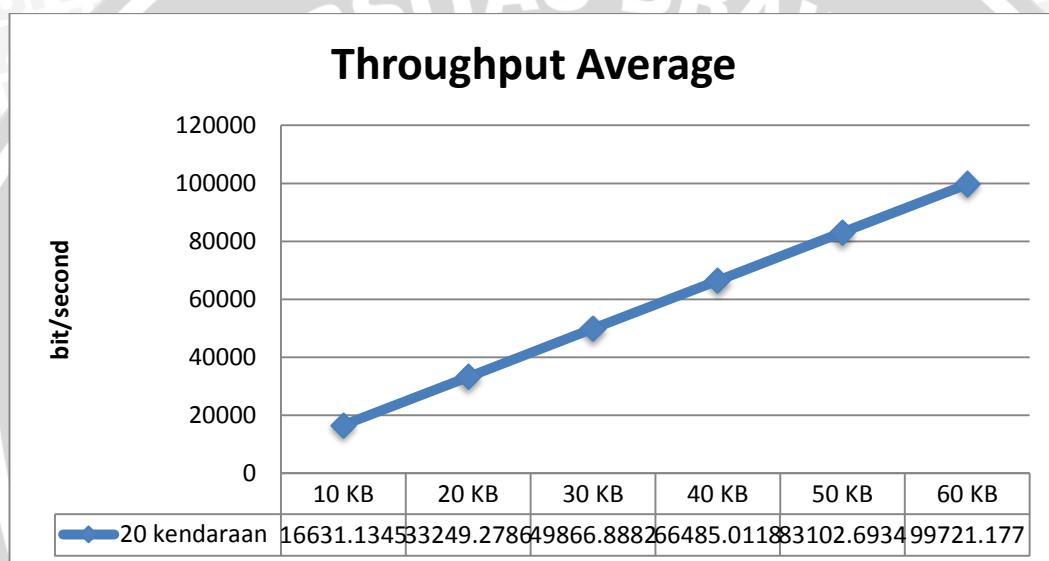
Gambar 5.3 Packet loss dari pengujian kepadatan

Gambar 5.3 merupakan hasil total *packet loss* dari hasil pengujian terhadap parameter kepadatan. Dari gambar grafik dapat dilihat pada pengujian bahwa semakin banyak jumlah kendaraan menimbulkan *packet loss* yang

semakin tinggi. Namun pada pengujian terjadi anomaly pengujian pada jumlah kendaraan 50 menjadi lebih rendah daripada jumlah kendaraan sebelumnya yaitu 10, 20, 30 dan 40 dikarenakan tipe *mobility* yang digunakan pada skenario pengujian ini adalah *random mobility*. Pengaruh jumlah kendaraan terhadap kinerja parameter terjadi dikarenakan *channel* yang digunakan untuk pengiriman data harus terbagi pula untuk setiap kendaraan yang menerima data sehingga terjadi pengaruh terhadap nilai dari hasil *packet loss*.

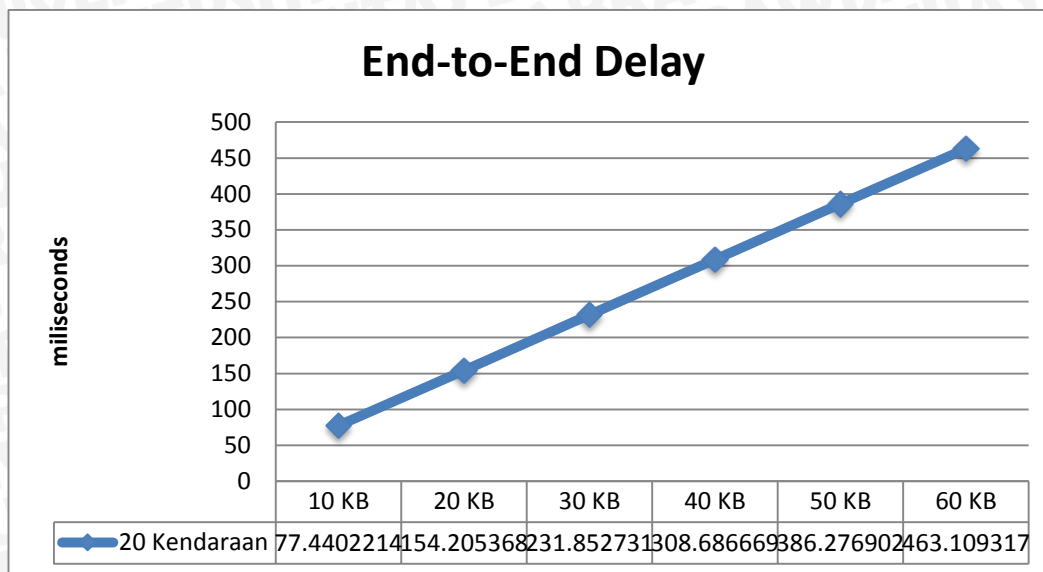
5.1.2 Pengujian Terhadap Parameter Besar Data

Pada sub bab ini terdapat hasil pengujian dari simulasi pengujian pengaruh besar data pada parameter kinerja dalam vanet. Pada Pengujian terdapat tiga parameter yang diuji yaitu *Throughput*, *End-to-End Delay* dan *Packet loss*.



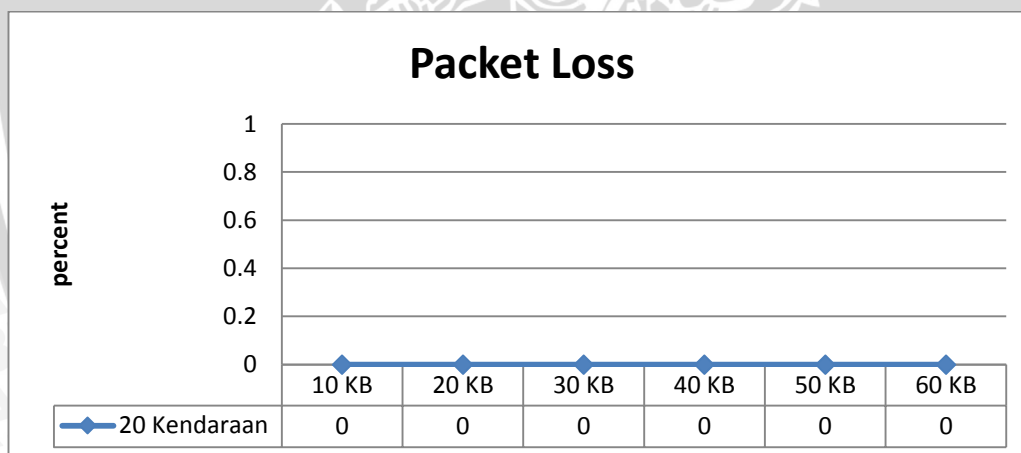
Gambar 5.4 Hasil Throughput pada Pengujian Besar Data

Gambar 5.4 merupakan hasil *throughput* pada pengujian besar data. Pada gambar dapat dilihat bahwa semakin besar data yang dikirimkan maka semakin besar pula nilai *throughput* yang didapatkan. Hal tersebut dikarenakan besar data yang dikirimkan tidak melebihi kemampuan dari *bandwidth* dan semakin besar data yang dikirimkan akan menyebabkan waktu pengiriman yang semakin lama pula.



Gambar 5.5 Delay pada pengujian besar data

Gambar 5.5 merupakan hasil *End-to-End Delay* pada pengujian besar data yang dikirimkan. Berdasarkan gambar dapat diketahui bahwa semakin besar data maka semakin tinggi *delay* yang dihasilkan. Jadi, dapat dikatakan bahwa proses transmisi berbanding lurus dengan besar data, sehingga semakin besar data semakin besar pula *end-to-end delay* yang dihasilkan.



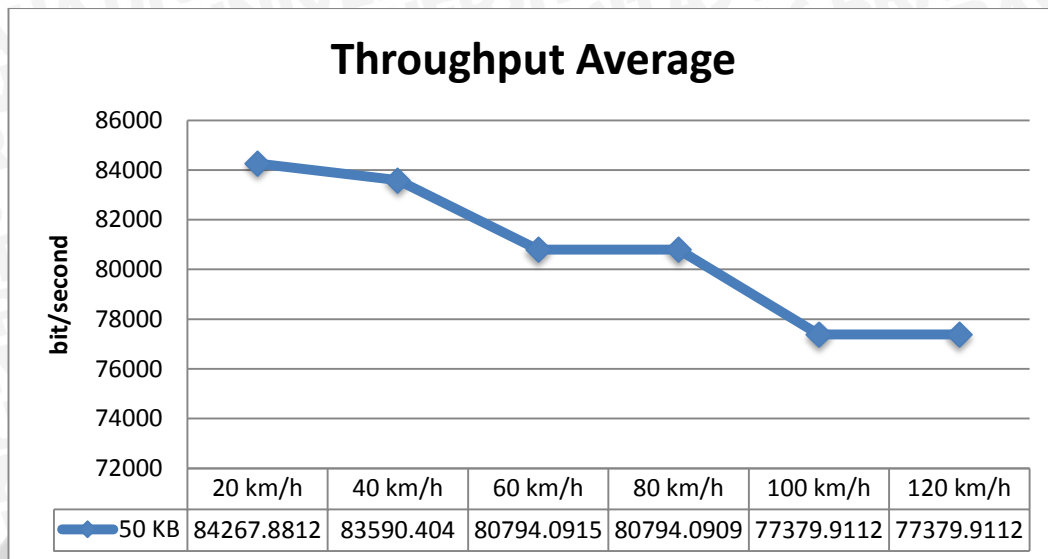
Gambar 5.6 Hasil Packet loss pada Pengujian Besar Data

Gambar 5.6 merupakan hasil total *Packet loss* pada pengujian besar data. Pada hasil pengujian dapat dilihat bahwa tidak terjadi *packet loss* dalam pengujian. Hal tersebut terjadi karena besarnya data yang dikirimkan masih dapat diterima oleh node yang bergerak dikarenakan jarak antar node tetap stabil dan masih dalam jangkauan pengiriman data serta waktu simulasi yang masih memungkinkan untuk melakukan pengiriman data.

5.1.3 Pengujian Terhadap Parameter Kecepatan

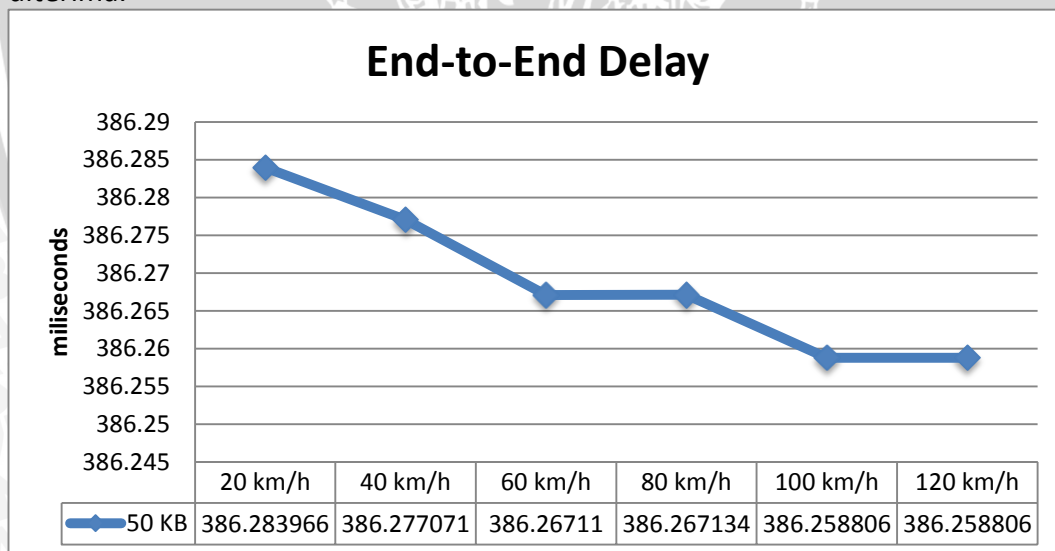
Pada sub bab ini terdapat hasil pengujian dari simulasi pengujian pengaruh kecepatan kendaraan pada parameter kinerja dalam vanet. Pada

Pengujian terdapat tiga parameter yang diuji yaitu *Throughput*, *End-to-End Delay* dan *Packet loss*.



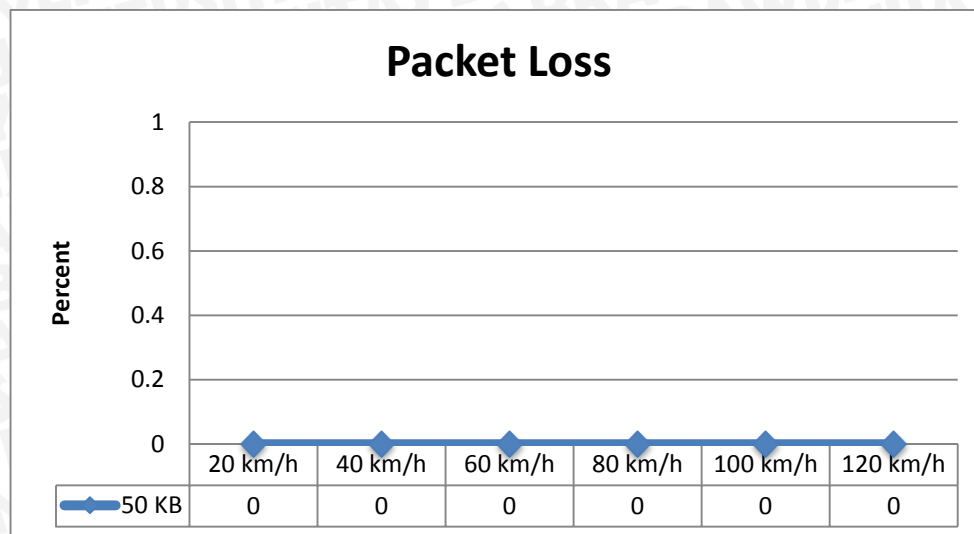
Gambar 5.7 Hasil Troughput pada pengujian kecepatan kendaraan

Gambar 5.7 merupakan hasil *throughput* pada pengujian dengan parameter kecepatan kendaraan. Pada hasil pengujian dapat dilihat bahwa hasil *throughput* menurun pada saat kecepatan kendaraan meningkat dikarenakan berpengaruh kuat pada waktu pengiriman suatu data yang akan semakin lama diterima.



Gambar 5.8 Hasil *End-to-End Delay* pada pengujian kecepatan kendaraan

Gambar 5.8 merupakan hasil *End-to-End Delay* pada pengujian dengan parameter kecepatan kendaraan. Pada hasil pengujian dapat dilihat bahwa hasil *delay* menurun pada saat kecepatan kendaraan meningkat. Hal ini dikarenakan bahwa semakin jauh jarak antar node semakin besar pula *delay* yang dihasilkan, dan begitu juga sebaliknya.



Gambar 5.9 Hasil Packet loss pada pengujian kecepatan kendaraan

Gambar 5.9 merupakan hasil *packet loss* pada pengujian dengan parameter kecepatan kendaraan. Pada hasil pengujian dapat dilihat bahwa tidak terjadi *packet loss* dalam pengujian. Hal tersebut terjadi karena besarnya data yang dikirimkan masih dapat diterima oleh node yang bergerak dikarenakan jarak antar node tetap stabil dan masih dalam jangkauan pengiriman data serta waktu simulasi yang masih memungkinkan untuk melakukan pengiriman data.

5.2 Analisis

Pada sub bab ini akan memberikan analisis dari setiap pengujian yang dilakukan. Analisis dilakukan dengan didasarkan pada data hasil pengujian. Analisis memperhatikan pengaruh dari parameter pengujian terhadap parameter kinerja.

5.2.1 Analisis Pengujian Terhadap Pengaruh Kepadatan

Pada bagian ini akan dijelaskan pengaruh kepadatan pada parameter kinerja. Berdasarkan hasil dari pengujian dapat diketahui bahwa variasi jumlah kendaraan mempengaruhi besar *throughput*, *End-to-End delay* dan *packet loss*.

Pada pengujian kepadatan terjadi kenaikan *throughput* secara bertahap sesuai dengan jumlah kendaraan pada simulasi. Pada hasil *End-to-End Delay* dapat dilihat terjadi kenaikan secara berkala pada setiap peningkatan kendaraan, sedangkan *Packet loss* meningkat setiap jumlah kendaraan meningkat. Peningkatan *delay* dikarenakan besarnya *Packet loss* menyebabkan penghitungan pada *delay* menjadi meningkat pula. Sehingga dari analisis ini dapat dikatakan peningkatan jumlah kendaraan mempengaruhi peningkatan *throughput*, peningkatan *End-to-End delay* dan peningkatan *packet loss*.

Pengaruh jumlah kendaraan terhadap kinerja parameter terjadi dikarenakan *channel* yang digunakan untuk pengiriman data harus terbagi untuk setiap kendaraan yang menerima data sehingga terjadi pengaruh terhadap *throughput*, *End-to-End Delay* dan *packet loss*.

5.2.2 Analisis Pengujian Terhadap Pengaruh Besar Data

Pada bagian ini akan dijelaskan pengaruh kepadatan pada parameter kinerja. Berdasarkan hasil dari pengujian dapat diketahui bahwa variasi besar data memberikan pengaruh pada *throughput*, *End-to-End delay* dan *packet loss*.

Pada hasil *throughput* dapat dilihat bahwa besarnya naik jika besar data semakin besar. Hal tersebut di karenakan data yang dikirimkan tidak melebihi kemampuan dari *bandwidth* dan semakin besar data yang dikirimkan akan menyebabkan waktu pengiriman yang semakin lama pula.

Pada hasil pengujian *End-to-End Delay* besarnya semakin naik setiap besar data yang dikirimkan bertambah. Berdasarkan rumus *delay* transmisi yaitu:

$$DT = \frac{N}{R}$$

Keterangan:

DT = *Delay* Transmisi

N = Besar Data yang Dikirim (Bytes)

R = *Bitrate* (bps)

Dapat dikatakan bahwa proses transmisi berbanding lurus dengan besar data, sehingga semakin besar data semakin besar *delay* yang dihasilkan.

Pada hasil pengujian semakin besar data tidak mempengaruhi hasil dari *packet loss* yang dihasilkan. Hal tersebut terjadi karena besarnya data yang dikirimkan masih dapat diterima oleh node yang bergerak dikarenakan jarak antar node tetap stabil dan masih dalam jangkauan pengiriman data serta waktu simulasi yang masih memungkinkan untuk melakukan pengiriman data.

5.2.3 Analisis Pengujian Terhadap Pengaruh Kecepatan

Pada bagian ini akan dijelaskan pengaruh kecepatan kendaraan pada parameter kinerja. Berdasarkan hasil dari pengujian dapat diketahui bahwa semakin cepat kecepatan kendaraan menyebabkan terjadinya penurunan terhadap *throughput*. Seiring meningkatnya kecepatan kendaraan, pengiriman suatu data akan semakin lama didapatkan.

Pada hasil pengujian dapat diketahui pula hasil *End-to-End Delay* semakin turun seiring meningkatnya kecepatan kendaraan. Hal ini dikarenakan bahwa semakin jauh jarak antar node semakin besar pula *delay* yang dihasilkan, dan begitu juga sebaliknya.

Pada hasil pengujian semakin meningkatnya kecepatan kendaraan tidak mempengaruhi hasil dari *packet loss* yang dihasilkan. Hal tersebut terjadi karena besarnya data yang dikirimkan masih dapat diterima oleh node yang bergerak dikarenakan jarak antar node tetap stabil dan masih dalam jangkauan pengiriman data serta waktu simulasi yang masih memungkinkan untuk melakukan pengiriman data.

BAB 6 PENUTUP

Bagian ini memuat kesimpulan dan saran terhadap skripsi. Kesimpulan dan saran disajikan secara terpisah, dengan penjelasan sebagai berikut:

6.1 Kesimpulan

Dari hasil implementasi, pengujian dan analisis dari simulasi komunikasi data antar kendaraan dengan menggunakan protokol 802.11b, dapat disimpulkan bahwa :

1. Untuk melakukan simulasi untuk komunikasi antar kendaraan diperlukan simulator OMNeT++ sebagai *network simulator* yang dilengkapi dengan *framework* INET dan SUMO sebagai *traffic simulator*. Setiap node pada simulasi memiliki data rate sebesar 11 Mbps, Tx Power 2mW dengan frekuensi 2.4 GHz.
2. Pengukuran performansi protokol 802.11b dapat dilakukan dengan menggunakan parameter *throughput*, *end-to-end delay* dan *packet loss* untuk menentukan kualitas layanan dari sebuah jaringan VANET.
3. Parameter uji kepadatan kendaraan, peningkatan kepadatan kendaraan akan mempengaruhi peningkatan *throughput*, peningkatan *delay* dan peningkatan *packet loss* dikarenakan *channel* yang digunakan terbagi untuk setiap kendaraan yang menerima data.
4. Parameter uji besar data yang ditransmisikan mempengaruhi *throughput* dan *End-to-End Delay* dikarenakan terjadinya peningkatan *delay* transmisi setiap data diperbesar.
5. Parameter uji kecepatan, setiap terdapat peningkatan kecepatan kendaraan akan menimbulkan penurunan *throughput* dan penurunan *delay*.
6. Pada skenario kepadatan lalu lintas memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 95942.9256 bps, 347.1401 ms dan 0.04 %. Pada skenario besar pengiriman data memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 56509.3635 bps, 270.256 ms dan 0 %. Sedangkan pada skenario kecepatan kendaraan memiliki rata – rata *throughput*, *End-to-End Delay* dan *packet loss* secara berurutan yakni 80701.04 bps, 386.26 ms dan 0 %.

6.2 Saran

Saran yang dapat disampaikan penulis untuk simulasi dan analisis komunikasi data pada VANET dengan menggunakan protokol 802.11b adalah :

1. Perlu dilakukan penelitian yang lebih lanjut untuk mengetahui kinerja protokol terhadap analisis parameter metrik yang digunakan, tidak terbatas pada skenario yang diterapkan pada penelitian ini.
2. Perlu dilakukan penelitian dengan melakukan implementasi yang nyata untuk mendapatkan hasil yang lebih konkret.

DAFTAR PUSTAKA

- Andras Varga and OpenSim Ltd, 2014. *Omnetpp*. Tersedia di: omnetpp.org/doc/omnetpp/manual/usman.html [Diakses 29 Juli 2015]
- Ekka, A., 2013. *A Simulation Of Vehicular Movement in VANET*. National Institute of Technology Rourkela. India.
- Institute of Transportation Systems, n.d. *SUMO – Simulation of Urban Mobility*. Tersedia di: <http://www.dlr.de/> [Diakses 27 Juli 2015]
- Karyono, Martoyo, I. & Uranus, H. P., 2008. *Perbandingan Simulator untuk Simulasi Wireless Sensor Network*. Universitas Multimedia Nusantara. Tangerang.
- Weiner, S. J., 2010. *Feasibility of a 802.11 VANET Based Car Accident Alert System*. Computer Engineering Northeastern University. Boston, Massachusetts.
- Kumar, R. & Dave, M., 2012. *A Review of Various VANET Data Dissemination Protocols*.
- Phule, H., 2012. *Public safety application for approaching emergency vehicle alert and accident reporting in VANET's using WAVE*. Rochester Institute of Technology. New York.
- Gunawan, E. K., Sugita, F. M., Abdurohim, M. W., 2012. *IEEE 802.11b*. Teknik Elektro Universitas Gadjah Mada. Yogyakarta.
- INET Framework. *What is INET Framework?*. Tersedia di: <https://inet.omnetpp.org/> [Diakses 3 Agustus 2015]
- Prokop, M. J., 2011. *Routing Protovol Evaluation and Development Of A Fully Functional Simulation Environment For Vehicular Ad Hoc Networks*. Rochester Institute of Technology. New York.
- Edgewall Software. 2003. *JOSM*. Tersedia di: <http://josm.openstreetmap.de/> [Diakses 18 Oktober 2015]
- Pearson, B., 2000. *Complementary Code Keying Made Simple*. Intersil Corporation.
- Firmansyah, A., 2008. *Teknik Spread Spectrum [lanjutan...]*. Tersedia di: <https://firmansyah2308.wordpress.com/tag/dsss/> [Diakses 12 November 2015]
- Muslim, M. A., 2007. *Analisis Codec dan Payload pada Micronet dan CISCO Pada Jaringan VPN-MPLS*. Fakultas Teknologi Informasi, Universitas Stikubank. Semarang.
- Dewo, E. S., 2003. *Bandwidth dan Throughput*. IlmuKomputer.Com.
- Rizaldi et al. 2010. *QoS (Quality of Service)*. Teknik Informatika Universitas Islam Negeri Sunan Kalijaga. Yogyakarta.

Radino, N., 2013. *Keunggulan dan kelemahan standar Wireless LAN*. Tersedia di:
<<http://nikolasa3.blogspot.co.id/2013/01/keunggulan-dan-kelemahan-standar.html>> [Diakses 12 November 2015]

