



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE 141599

**EVALUASI KOMUNIKASI KOOPERATIF PADA SISTEM
*SMART LIGHTING***

Muhammad Ilham Pratomo
NRP 07111440000167

Dosen Pembimbing
Dr. Ir. Endroyono, DEA
Ir. Gatot Kusrahardjo, MT.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

**EVALUATION OF COOPERATIVE COMMUNICATION
IN SMART LIGHTING SYSTEM**

Muhammad Ilham Pratomo
NRP 07111440000167

Supervisors
Dr. Ir. Endroyono, DEA
Ir. Gatot Kusrahardjo, MT.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2018

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan tugas akhir saya dengan judul “**Evaluasi Komunikasi Kooperatif pada Sistem *Smart Lighting***” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2018

Muhammad Ilham Pratomo
NRP. 07111440000167

**EVALUASI KOMUNIKASI KOOPERATIF PADA
SISTEM *SMART LIGHTING***

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik Elektro
Pada**

**Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui

Dosen Pembimbing I,

Dosen Pembimbing II,

Dr. Ir. Endroyono, DEA NIP. 196504041991021001

Gatot Kusrahardjo, MT. NIP. 195904281986011001



EVALUASI KOMUNIKASI KOOPERATIF PADA SISTEM *SMART LIGHTING*

Muhammad Ilham Pratomo
07111440000167

Dosen Pembimbing I : Dr. Ir. Endroyono, DEA
Dosen Pembimbing II : Ir. Gatot Kusrahardjo, MT.

ABSTRAK

Smart lighting system merupakan salah satu sistem yang diterapkan pada salah satu pilar pada *smart city* yang berfungsi untuk mempermudah masyarakat maupun pemerintah kota dalam mengontrol sistem pencahayaan pada penerangan kota dan memonitoring kondisi yang terjadi pada lingkungan umum. Sistem *smart lighting* pada umumnya menerapkan sistem komunikasi kooperatif. Sistem ini diterapkan karena kestabilannya dalam hal efisiensi energi karena mengurangi daya transmisi serta dapat mengoptimalkan kapasitas data yang akan ditransmisikan. Kecerdasan pada kemampuan *smart lighting* bergantung pada sistem yang dibutuhkan suatu kota dalam menyelesaikan suatu permasalahan. Semakin banyak permasalahan yang ingin diatasi maka dibutuhkan kemampuan pada *smart lighting* yang semakin lengkap. Kemampuan ini bergantung pada jumlah sensor yang digunakan. Oleh sebab itu pada tugas akhir ini akan dilakukan evaluasi kinerja pada sistem komunikasi kooperatif yang diterapkan pada sistem *smart lighting* menggunakan *software Network Simulator 2*. Sensor yang digunakan pada sistem ini adalah sensor suhu, kelembapan, cahaya, arus listrik dan PIR. Parameter yang akan di evaluasi adalah kapasitas data yang terdiri dari *throughput* dan *delay* serta jumlah terminal dan jarak antar terminal pada *smart lighting*. Dari hasil penelitian didapatkan jarak antar terminal dengan nilai QoS tertinggi, nilai dari jumlah *sent packet*, *loss packet*, *dropped packet* dan nilai dari *end-to-end delay*. Nilai dari jumlah paket akan semakin meningkat dengan bertambahnya jumlah terminal pada *smart lighting*.

Kata Kunci : *Smart Lighting*, Sistem Komunikasi Kooperatif, *Network Simulator 2*, *Smart City*

[Halaman ini sengaja dikosongkan]

EVALUATION OF COOPERATIVE COMMUNICATION IN SMART LIGHTING SYSTEM

Muhammad Ilham Pratomo
07111440000167

Supervisor I : Dr. Ir. Endroyono, DEA
Supervisor II : Ir. Gatot Kusrahardjo, MT.

ABSTRACT

Smart lighting system is one of the systems applied to one of the pillars of smart city that serves to facilitate the public and city government in controlling the lighting system in the city and monitoring conditions that occur in the general environment. Smart lighting system generally apply cooperative communication system. This system is applied because of its stability in terms of energy efficiency because it reduces transmission power and can optimize the capacity of data to be transmitted. Intelligence in the ability of smart lighting depends on the system needed in solving a problem that exist in the city. The more problems in the city want to solved the more intelligent capabilities required in the more complete. This capability depends on the number of sensors used. The more sensors used, the more intelligent the ability of the smart lighting and the more sensors used, the greater the amount of data that must be transmitted, thus requiring large capacity. Therefore in this final task will be performed performance evaluation of cooperative communication system applied to smart lighting system using Network Simulator 2. System of smart lighting using a few sensors like temperature sensors, humidity, light, electric current and PIR. The parameters to be evaluated is the data capacity consisting of throughput & delay and the number of terminals and the distance between terminals on smart lighting. From the research results obtained distance between the terminal with the highest QoS value, the value of the number of sent packets, packet loss, dropped packet and the value of end-to-end delay. The value of the number of packets will increase with the increasing number of terminals in the smart lighting.

Keyword : *Smart Lighting*, Communication cooperative, *Network Simulator 2*, *Smart City*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Alhamdulillah, puji syukur penulis ucapkan kehadiran Illahi Rabbi, Allah SWT atas segenap karunia-Nya yang tak terhitung jumlahnya, sehingga penulis dapat menyelesaikan penulisan buku Tugas Akhir dengan judul “**Evaluasi Komunikasi Kooperatif pada Sistem Smart Lighting**”. Tugas akhir merupakan salah satu syarat yang harus dipenuhi untuk menyelesaikan program studi Strata-1 pada Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya. Penulis menyadari bahwa tugas akhir ini adalah bagian dari proses pematangan kapasitas akademik dalam banyak hal.

Pada kesempatan ini penulis ingin menyampaikan ucapan terimakasih dan penghargaan setinggi-tingginya kepada :

1. Kedua orang tua penulis, yang selalu memberikan dukungan finansial maupun moral selama penulis menjalankan proses perkuliahan di ITS, sampai akhirnya bisa menyelesaikan tugas akhir ini.
2. Bapak Dr. Ir. Endroyono, DEA. dan Ir. Gatot Kusrahardjo, MT. selaku Dosen Pembimbing atas arahan dan bimbingannya selama pengerjaan tugas akhir ini.
3. Bapak dan Ibu dosen departemen teknik elektro ITS, khususnya bidang studi Telekomunikasi Multimedia yang telah mendidik dan memberikan ilmunya dari awal perkuliahan sampai bisa menyelesaikan tugas akhir ini.
4. Semua rekan-rekan di lab b304 komunikasi multimedia yang telah saling bekerja dan belajar bersama selama mengerjakan Tugas Akhir ini.
5. Teman-teman Elektro ITS angkatan 2014, yang telah menjadi sahabat dan teman seperjuangan selama penulis berkuliah 4 tahun di surabaya.
6. Teman-teman Kalpataru Elektro ITS yang telah menjadi rumah dan keluarga bagi penulis selama berkuliah di Elektro ITS surabaya.

Penulis menyadari bahwa dalam tugas akhir ini masih belum sempurna. Semoga ketidak sempurnaan itu menjadi celah untuk proses belajar dan pengembangan rancangan kedepannya. Semoga Tugas Akhir ini bisa menjadi bahan referensi dan memberikan manfaat bagi pengembangan dunia telekomunikasi.

Penulis berharap Buku Tugas Akhir ini dapat berguna untuk seluruh pembaca dan khususnya civitas departemen teknik elektro. Terlebih lagi, diharapkan buku ini dapat menjadi acuan dalam hal kemajuan teknologi di Indonesia.

Surabaya, Mei 2018

Penulis

DAFTAR ISI

1	JUDUL	i
2	PERNYATAAN KEASLIAN.....	v
3	LEMBAR PENGESAHAN....	Error! Bookmark not defined.
4	ABSTRAK.....	ix
5	ABSTRACT	xi
6	KATA PENGANTAR	xiii
7	DAFTAR ISI	xv
8	DAFTAR GAMBAR	xix
9	DAFTAR TABEL	xxi
1	BAB 1 PENDAHULUAN.....	1
	1.1. Latar Belakang	1
	1.2. Rumusan Masalah	2
	1.3. Batasan Masalah	2
	1.4. Tujuan	2
	1.5. Metodologi.....	2
	1.6. Sistematika Pembahasan	3
	1.7. Relevansi	4
2	BAB 2 TINJAUAN PUSTAKA.....	5
	2.1 Penerangan Jalan Umum (PJU)	5
	2.1.1 Dasar Perencanaan Penerangan Jalanm	5
	2.1.2 Jenis Lampu Penerangan Jalan	6
	2.1.3 Jarak Antar Lampu Penerangan Jalan.....	7
	2.2 Smart Lighting	8

2.2.1	Smart Lighting System	10
2.3	Sistem Komunikasi Kooperatif.....	11
2.4	Sensor	11
2.4.1	Sensor Suhu dan Kelembapan	12
2.4.2	Sensor Intensitas Cahaya.....	13
2.4.3	Sensor <i>Infrared</i>	14
2.4.4	Sensor Arus Listrik	15
2.5	Protokol Komunikasi	16
2.5.1	Wi-Fi	17
2.5.2	Zigbee	17
2.6	Estimasi Kapasitas Bandwidth	18
2.7	Analisa Propagasi	19
2.7.1	Free Space Path Loss	19
2.7.2	Propagasi Two-Ray Ground	20
2.7.3	Propagasi Shadowing	21
2.8	Delay (Latency).....	21
2.9	Throughput	22
2.10	<i>Packet Loss Ratio</i>	22
2.11	Network Simulator-2 (NS-2)	23
2.11.1	Protokol dan Model Pada NS-2	25
2.11.2	Transport Agent pada NS-2	25
2.11.3	Level Aplikasi pada NS-2.....	26
3	BAB 3 PERANCANGAN DAN SIMULASI SISTEM.....	27
3.1	Diagram Alir Perancangan	27
3.2	Kajian Lingkungan Penerangan Jalan Umum.....	28
3.3	Penentuan Parameter Simulasi	29
3.3.1	Perhitungan Model Propagasi dan Link Budget	31
3.4	Desain Simulasi pada NS-2.....	32
3.4.1	Pembangkitan Node	33
3.4.2	Koneksi Lapisan Transport dan Aplikasi	35

3.5	Skenario Simulasi	37
3.5.1	Pembuatan Topologi Jaringan	38
3.5.1.1	Topologi Jaringan dengan 3 node	38
3.5.1.2	Topologi Jaringan dengan 5 node	39
3.5.1.3	Topologi Jaringan dengan 7 node	40
3.5.1.4	Topologi Jaringan dengan 9 node	40
3.5.1.5	Topologi Jaringan <i>Point-to-Multipoint</i>	41
4	BAB 4	43
4.1	Pendahuluan	43
4.2	Performa Komunikasi Kooperatif.....	43
4.2.1	Performa Pada Jarak 30 Meter.....	43
4.2.1.1	Performa Sistem Pada Topologi 3 Node	44
4.2.1.2	Performa Sistem Pada Topologi 5 Node	44
4.2.1.3	Performa Sistem Pada Topologi 7 Node	45
4.2.1.4	Performa Sistem Pada Topologi 9 Node	46
4.2.2	Performa Pada Jarak 40 Meter.....	47
4.2.2.1	Performa Sistem Pada Topologi 3 Node	47
4.2.2.2	Performa Sistem Pada Topologi 5 Node	48
4.2.2.3	Performa Sistem Pada Topologi 7 Node	49
4.2.2.4	Performa Sistem Pada Topologi 9 Node	50
4.2.3	Performa Pada Jarak 50 Meter.....	51
4.2.3.1	Performa Sistem Pada Topologi 3 Node	52
4.2.3.2	Performa Sistem Pada Topologi 5 Node	53
4.2.3.3	Performa Sistem Pada Topologi 7 Node	54
4.2.3.4	Performa Sistem Pada Topologi 9 Node	55
4.3	Analisa Performa Pada Topologi <i>Point-to-Point</i>	56
4.3.1	Analisa Performa Jaringan pada jarak 30 Meter	56
4.3.2	Analisa Performa Jaringan pada jarak 40 Meter	58
4.3.3	Analisa Performa Jaringan pada jarak 50 Meter	59
4.4	Perbandingan Performa Jaringan pada Seluruh Topologi <i>Point-to-Point</i>	61
4.5	Hasil Throughput	63

4.6	Performa Routing Jaringan Kooperatif.....	64
BAB 5 KESIMPULAN DAN SARAN		67
5.1	Kesimpulan	67
5.2	Saran.....	68
6	DAFTAR PUSTAKA.....	69
7	LAMPIRAN.....	71
8	BIOGRAFI PENULIS.....	115

DAFTAR GAMBAR

Gambar 2.1 Smart Lighting	9
Gambar 2.2 Sistem <i>Smart Lighting</i>	10
Gambar 2.3 Komunikasi Kooperatif pada Smart Lighting [6]	11
Gambar 2.4 Sensor DHT22	12
Gambar 2.5 Sensor LDR	13
Gambar 2.6 Sensor Arus Listrik	15
Gambar 2.7 Ilustrasi Free Space Path Loss	20
Gambar 2.8 Nilai <i>Packet Loss Ratio</i> (%)	23
Gambar 2.9 Arsitektur NS-2	24
Gambar 2.10 Tampilan NAM pada NS-2	25
Gambar 3.1 Diagram Alir Perancangan Simulasi	27
Gambar 3.2 Frame struktur dalam satu paket data	30
Gambar 3.3 Model Sistem Menggunakan NS-2	33
Gambar 3.4 Kode Pembangunan <i>Node</i>	34
Gambar 3.5 Kode Pengaturan Parameter <i>Node</i>	35
Gambar 3.6 Kode Pengaturan Agen <i>Transport</i>	36
Gambar 3.7 Pengaturan <i>Application Layer</i>	37
Gambar 3.8 Topologi Jaringan 3 <i>Node</i>	39
Gambar 3.9 Topologi Jaringan 5 <i>Node</i>	39
Gambar 3.10 Topologi Jaringan 7 <i>Node</i>	40
Gambar 3.11 Topologi Jaringan 9 <i>Node</i>	40
Gambar 3.12 Topologi Jaringan <i>Point-to-Multipoint</i>	41
Gambar 4.1 Performa Jaringan Kooperatif pada Jarak 30 Meter	57
Gambar 4.2 Nilai <i>End-to-end Delay</i> pada Jarak 30 Meter	57
Gambar 4.3 Performa Jaringan Kooperatif pada Jarak 40 Meter	58
Gambar 4.4 Nilai <i>End-to-end Delay</i> pada Jarak 40 Meter	59
Gambar 4.5 Performa Jaringan Kooperatif pada Jarak 50 Meter	60
Gambar 4.6 Performa Jaringan Kooperatif pada Jarak 50 Meter	61
Gambar 4.7 Performa Jaringan pada Topologi <i>Point-to-Point</i>	61
Gambar 4.8 Perbandingan <i>End-to-end Delay</i>	62
Gambar 4.9 Hasil Throughput pada Jarak Antar <i>Node</i> 50 meter	63

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 2.1 Jenis Lampu Penerangan Jalan Umum	6
Tabel 2.2 Jarak Antar Lampu tipe A	7
Tabel 2.3 Jarak Antar Lampu Tipe B	8
Tabel 2.4 Perbandingan Standar Komunikasi	16
Tabel 2.5 Parameter Path Loss Exponent dan Shadowing Deviation	21
Tabel 3.1 Parameter Simulasi pada NS-2	29
Tabel 3.2 Parameter Propagasi pada NS-2	32
Tabel 4.1 Informasi Simulasi pada Topologi 3 Node (30 meter)	44
Tabel 4.2 Informasi Simulasi pada Topologi 5 Node (30 meter)	45
Tabel 4.3 Informasi Simulasi pada Topologi 7 Node (30 meter)	46
Tabel 4.4 Informasi Simulasi pada Topologi 9 Node (30 meter)	47
Tabel 4.5 Informasi Simulasi pada Topologi 3 Node (40 meter)	48
Tabel 4.6 Informasi Simulasi pada Topologi 5 Node (40 meter)	49
Tabel 4.7 Informasi Simulasi pada Topologi 7 Node (40 meter)	50
Tabel 4.8 Informasi Simulasi pada Topologi 9 Node (40 meter)	51
Tabel 4.9 Informasi Simulasi pada Topologi 3 Node (50 meter)	52
Tabel 4.10 Informasi Simulasi pada Topologi 5 Node (50 meter)	53
Tabel 4.11 Informasi Simulasi pada Topologi 7 Node (50 meter)	54
Tabel 4.12 Informasi Simulasi pada Topologi 9 Node (50 meter)	55
Tabel 4.13 Performa Jaringan Kooperatif Pada Jarak 30 Meter	56
Tabel 4.14 Performa Jaringan Pada Jarak 40 Meter	58
Tabel 4.15 Performa Jaringan Pada Jarak 50 Meter	59
Tabel 4.16 Informasi Simulasi pada Topologi Point-to-Multipoint	64
Tabel 4.17 Informasi <i>Routing Table</i> dari Setiap <i>Node</i>	65

[Halaman ini sengaja dikosongkan]

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Saat ini perkembangan dari *smart city* sedang marak dikembangkan pada setiap kota di dunia. Hampir dari setiap pemerintah kota di dunia ingin mengembangkan konsep dan sistem *smart city* guna membantu berbagai hal kegiatan untuk masyarakatnya, terutama dalam upaya mengelola sumber daya yang ada dengan efektif, memberikan kenyamanan dan keamanan serta memberikan kemudahan mengakses informasi bagi masyarakat, hingga untuk mengantisipasi kejadian yang tak terduga sebelumnya. Pada *Smart City* terdiri dari beberapa aspek yang dikembangkan, salah satunya adalah fasilitas umum untuk melayani kebutuhan masyarakat. Fasilitas umum yang banyak digunakan adalah sistem penerangan pada lingkungan kota. Sistem penerangan ini berupa lampu jalan yang memiliki kemampuan cerdas seperti sistem otomatis yang dapat mengatur nyala atau hidupnya lampu, akses layanan internet berupa wifi, informasi mengenai kondisi polusi udara hingga kondisi kemacetan pada jalan-jalan di dalam kota. Sistem penerangan ini biasa disebut dengan istilah *smart lighting*.

Smart lighting dapat mengatasi permasalahan serta mempermudah sistem yang ada untuk saat ini, Tetapi ada beberapa aspek penting yang perlu dipertimbangkan dalam mengimplementasikan sistem tersebut. Aspek pertama berupa hubungan komunikasi antar lampu yang dapat digunakan. Hubungan komunikasi ini dapat menggunakan kabel *fiber optic* ataupun berupa *wireless*. Namun hubungan komunikasi antar lampu pada *smart lighting* pada umumnya menggunakan *wireless* dengan sistem komunikasi kooperatif. Sistem ini diterapkan agar sistem dapat bekerja secara efisien. sehingga diperlukan hubungan komunikasi berupa *access point* yang mampu mentransmisikan data antar terminal lampu pada *smart lighting*. Aspek kedua berupa sensor-sensor yang dapat mengatasi kondisi atau permasalahan yang ada pada lingkungan di sekitar *smart lighting*. Sensor tersebut dapat berupa sensor untuk mendeteksi polusi udara, sensor cahaya, sensor pendeteksi ketinggian air hujan hingga sensor yang dapat mengamati kondisi atau aktifitas pada jalan yaitu sensor kamera atau CCTV. Semakin banyak sensor yang digunakan maka semakin besar pula data yang perlu di transmisikan. Aspek ketiga adalah

pancaran daya yang di pancarkan oleh terminal pada *smart lighting*, karena hal ini yang akan menentukan jarak antar terminal dan keterbatasan kapasitas yang dapat di transmisikan.

Tidak sedikit pihak-pihak maupun perusahaan yang ingin menerapkan sistem komunikasi kooperatif pada sistem *smart lighting* namun tidak mempertimbangkan beberapa aspek tersebut. Oleh karena itu diperlukan evaluasi kinerja pada sistem komunikasi kooperatif pada sistem *smart lighting*, agar sistem tersebut dapat bekerja secara optimal tanpa adanya kesalahan pada sistem.

1.2. Rumusan Masalah

Permasalahan yang akan dibahas dalam tugas akhir ini adalah:

1. Berapa jumlah kapasitas data pada sistem *smart lighting* ?
2. Berapa jumlah terminal yang mampu melakukan komunikasi kooperatif pada sistem *smart lighting*?
3. Berapa jarak antar terminal pada sistem *smart lighting* ?

1.3. Batasan Masalah

Penelitian tugas akhir ini difokuskan pada komunikasi kooperatif antar terminal pada sistem *smart lighting*.

1.4. Tujuan

Penelitian tugas akhir ini bertujuan untuk mengetahui hasil kapasitas data dari seluruh terminal dan kinerja sistem tersebut agar sistem komunikasi kooperatif yang diterapkan pada *smart lighting* dapat bekerja tanpa adanya kesalahan pada sistem.

1.5. Metodologi

Metodologi yang digunakan dalam tugas akhir ini terdiri dari beberapa tahap sebagai berikut:

1. Studi Literatur

Mempelajari prinsip kerja dari sistem *smart lighting* khususnya pada sistem komunikasi kooperatif. Sistem komunikasi kooperatif ini yang akan mengirimkan data antara terminal pada *smart lighting*. Selain itu mempelajari cara menghitung kapasitas data pada suatu jaringan serta pengaruhnya terhadap daya pancar dan jarak antar terminal, agar kapasitas yang di implementasikan pada sistem *smart lighting* tersebut dapat di evaluasi.

2. Simulasi

Melakukan simulasi proses pengiriman data antar *smart lighting* melalui jaringan komunikasi kooperatif. Simulasi tersebut akan dibuat menggunakan software Network Simulator - 2 (NS-2). Parameter yang di inputkan pada simulasi tersebut berupa data pada setiap terminal, *gain* antenna, jarak antar terminal, dan lain-lain. Tujuan dari simulasi ini adalah mengetahui kinerja jaringan pada proses pengiriman informasi dari setiap *node* pada sistem *smart lighting*. Simulasi tersebut dilakukan dengan cara membuat beberapa *node* yang diasumsikan sebagai terminal pada *smart lighting* yang akan mengirimkan data melalui beberapa relay secara kooperatif menuju *destination*.

3. Analisa Data

Setelah simulasi telah selesai dilakukan maka akan dilakukan analisa terhadap hasil simulasi yang telah didapat. Setelah itu akan dianalisa berapa jarak antar terminal yang sesuai, jumlah kapasitas data yang mampu dikirimkan pada sistem *smart lighting*.

4. Kesimpulan

Pada akhir penelitian akan ditarik kesimpulan berdasarkan analisa yang telah dilakukan.

1.6. Sistematika Pembahasan

Pembahasan dalam tugas akhir ini akan dibagi dalam lima bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Pada bab ini dijelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi, sistematika pembahasan dan relevansi.

Bab II Tinjauan Pustaka

Pada bab ini dijelaskan mengenai tinjauan pustaka mengenai *Smart Lighting* baik dari sisi konsep secara umum, komponennya maupun sistem komunikasi yang digunakan pada sistem *Smart Lighting*.

Bab III Perancangan dan Simulasi Sistem

Pada bab ini dijelaskan hal-hal dan data-data yang berkaitan secara langsung dengan perancangan sistem *smart lighting*.

Bab IV Analisis Kinerja Sistem

Pada bab ini berisi data pengujian dari simulasi sistem yang telah dibuat untuk kemudian dianalisis.

Bab V Kesimpulan dan Saran

Pada bab ini berisi kesimpulan dan saran.

1.7. Relevansi

Tugas Akhir ini memberikan hasil jarak antar terminal, jumlah terminal dan jumlah kapasitas data yang ditransmisikan oleh seluruh terminal pada sistem komunikasi kooperatif pada sistem *Smart Lighting* agar sistem tersebut dapat diimplementasikan pada sistem *Smart City* di suatu kota tertentu.

BAB 2

TINJAUAN PUSTAKA

2.1 Penerangan Jalan Umum (PJU)

Lampu jalan atau biasa disebut dengan Penerangan Jalan Umum (PJU) adalah lampu yang digunakan untuk penerangan jalan di malam hari sehingga mempermudah pejalan kaki, pengguna sepeda dan pengendara kendaraan untuk melihat dengan lebih jelas jalan ataupun medan yang akan dilalui pada malam hari, sehingga dapat meningkatkan keselamatan lalu lintas dan keamanan dari para pengguna jalan dari kegiatan atau aksi kriminal. Penerangan Jalan umum memiliki beberapa fungsi antara lain:

1. Menghasilkan kontras antara obyek dan permukaan jalan
2. Sebagai alat bantu navigasi untuk pengguna jalan
3. Meningkatkan keselamatan dan kenyamanan bagi pengguna jalan, khususnya pada malam hari
4. Mendukung keamanan lingkungan
5. Memberikan keindahan lingkungan jalan

2.1.1 Dasar Perencanaan Penerangan Jalan

1) Perencanaan penerangan jalan terkait dengan hal-hal berikut ini :

- a) Volume lalu-lintas, baik kendaraan maupun lingkungan yang bersinggungan seperti pejalan kaki, pengayuh sepeda, dll
- b) Tipikal potongan melintang jalan, situasi (lay-out) jalan dan persimpangan jalan
- c) Geometri jalan, seperti alinyemen horisontal, alinyemen vertikal, dll;
- d) Tekstur perkerasan dan jenis perkerasan yang mempengaruhi pantulan cahaya lampu penerangan
- e) Pemilihan jenis dan kualitas sumber cahaya/lampu, data fotometrik lampu dan lokasi sumber listrik;
- f) Tingkat kebutuhan, biaya operasi, biaya pemeliharaan, dan lain-lain, agar perencanaan sistem lampu penerangan efektif dan ekonomis
- g) Rencana jangka panjang pengembangan jalan dan pengembangan daerah sekitarnya
- h) Data kecelakaan dan kerawanan di lokasi

2) Beberapa tempat yang memerlukan perhatian khusus dalam perencanaan penerangan jalan antara lain sebagai berikut :

- a) Lebar ruang milik jalan yang bervariasi dalam satu ruas jalan
- b) Tempat-tempat dimana kondisi lengkung horisontal (tikungan) tajam
- c) Tempat yang luas seperti persimpangan, *interchange* & tempat parkir
- d) Jalan-jalan berpohon
- e) Jalan-jalan dengan lebar median yang sempit, terutama untuk pemasangan lampu di bagian median
- f) Jembatan sempit/panjang, jalan layang dan jalan bawah tanah (terowongan)
- g) Tempat-tempat lain dimana lingkungan jalan banyak berinterferensi dengan jalannya.

2.1.2 Jenis Lampu Penerangan Jalan

Jenis lampu penerangan jalan ditinjau dari karakteristik dan penggunaannya secara umum dapat dilihat dalam tabel berikut.

Jenis lampu	Efisiensi rata-rata (lumen/watt)	Umur rencana rata-rata (jam)	Daya (watt)	Pengaruh ke warna obyek	keterangan
Lampu tabung <i>fluorescent</i> tekanan rendah	60 – 70	8.000 – 10.000	18 - 20; 36 - 40	Sedang	-jalan lokal -efisiensi tinggi -umur pendek
Lampu gas merkuri tekanan tinggi	50 – 55	16.000 – 24.000	125; 250; 400; 700	Sedang	-jalan lokal -efisiensi rendah -umur panjang
Lampu gas sodium bertekanan rendah	100 - 200	8.000 - 10.000	90; 180	Sangat buruk	-persimpangan -terowongan -efisiensi tinggi -umur panjang -cahaya buruk -dianjurkan karena efisiensi tinggi
Lampu gas sodium tekanan tinggi	110	12.000 - 20.000	150; 250; 400	Buruk	-jalan tol,arteri -efisiensi tinggi -umur panjang -pengontrolan cahaya mudah - <i>Recomendeed</i>

Tabel 2.1 Jenis Lampu Penerangan Jalan Umum

Tabel 2.1 menjelaskan mengenai jenis-jenis lampu yang digunakan pada Penerangan Jalan Umum kondisi saat ini. Jenis lampu tersebut memiliki intensitas cahaya yang berbeda, hal ini akan mempengaruhi jarak antar lampu pada penerangan jalan umum.

2.1.3 Jarak Antar Lampu Penerangan Jalan

Sesuai dengan Standar Nasional Indonesia, Penerangan Jalan Umum memiliki jarak antar tiang lampu penerangan berdasarkan tipikal distribusi pencahayaan dan klasifikasi lampu yang digunakan. Selain dari jenis lampu yang digunakan, tipe pada penerangan ini terdiri dari dua, yaitu rumah lampu tipe A dan rumah lampu tipe B. Rumah lampu (*lantern*) tipe A mempunyai penyebaran sorotan cahaya/sinar lebih luas. Sedangkan rumah lampu (*lantern*) tipe B mempunyai penyebaran sorotan cahaya yang lebih ringan/kecil, terutama yang langsung ke jalan. Tabel 2.2 dan tabel 2.3 akan menjelaskan mengenai jarak antar tiang lampu pada penerangan jalan umum berdasarkan tipe dari rumah lampu yang digunakan.

Jenis lampu	Tinggi lampu (m)	Lebar jalan (m)								Tingkat pencahayaan
		4	5	6	7	8	9	10	11	
35W SOX	4	32	32	32	-	-	-	-	-	3,5 LUX
	5	35	35	35	35	35	34	32	-	
	6	42	40	38	36	33	31	30	29	
55W SOX	6	42	40	38	36	33	32	30	28	6,0 LUX
90W SOX	8	60	60	58	55	52	50	48	46	
90W SOX	8	36	35	35	33	31	30	29	28	10,0 LUX
135W SOX	10	46	45	45	44	43	41	40	39	
135W SOX	10	-	-	25	24	23	22	21	20	20,0 LUX
180W SOX	10	-	-	37	36	35	33	32	31	
180W SOX	10	-	-	-	-	22	21	20	20	30,0 LUX

Tabel 2.2 Jarak Antar Lampu tipe A

Berdasarkan tabel 2.2 jarak antar lampu menyesuaikan dengan tingkat pencahayaan yang dihasilkan oleh jenis lampu serta tipe dari rumah lampu tersebut. Selain kedua hal tersebut tinggi dari lampu tersebut juga mempengaruhi dan lebar dari jalan pada lokasi penerangan jalan umum juga menjadi parameter untuk menentukan nilai dari jarak antar lampu pada rumah lampu tipe A.

Jenis lampu	Tinggi lampu (m)	Lebar jalan (m)									Tingkat pencahayaan
		4	5	6	7	8	9	10	11		
50W SON atau 80W MBF/U	4	31	30	29	28	26	-	-	-	3,5 LUX	
	5	33	32	32	31	30	29	28	27		
70W SON atau 125WMBF/U	6	48	47	46	44	43	41	39	37		
70W SON atau 125WMBF/U	6	34	33	32	31	30	28	26	24	6,0 LUX	
100W SON	6	48	47	45	42	40	38	36	34		
150W SON atau 250W MBF/U	8	-	-	48	47	45	43	41	39	10 LUX	
100W SON	6	-	-	28	26	23	-	-	-		
250W SON atau 400W MBF/U	10	-	-	-	-	55	53	50	47		
250W SON atau 400W MBF/U	10	-	-	36	35	33	32	30	28	20 LUX	
400W SON	12	-	-	-	-	39	38	37	36	30 LUX	

Tabel 2.3 Jarak Antar Lampu Tipe B

Pada rumah lampu Tipe B yang ditunjukkan pada tabel 2.3 nilai dari jarak antar lampu relatif lebih kecil dikarenakan penyebaran cahaya yang dapat dilakukan oleh rumah lampu tipe ini lebih kecil. Dari kedua tabel tersebut, terlihat bahwa semakin besar lebar jalan pada lokasi PJU maka jarak antar lampu semakin kecil. Hal ini disebabkan karena pada jalan yang lebih luas membutuhkan pencahayaan yang lebih besar.

2.2 Smart Lighting

Smart Lighting atau lampu cerdas merupakan lampu hemat energi yang digunakan untuk penerangan pada jalan umum yang memiliki

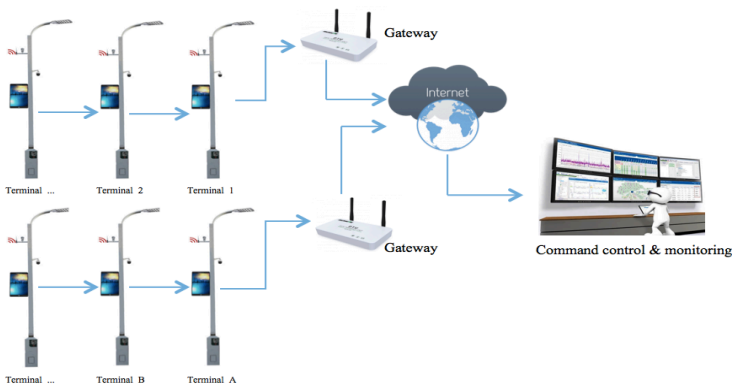
kemampuan cerdas yang dapat memberikan manfaat untuk masyarakat. Kemampuan cerdas ini bergantung pada fasilitas yang diberikan yang oleh lampu cerdas tersebut. Semakin lengkap fasilitas yang dapat dimanfaatkan dari lampu cerdas tersebut maka dapat dikatakan bahwa semakin cerdas lampu tersebut. Fasilitas ini terdiri dari beberapa sensor yang digunakan untuk memberikan beberapa informasi seperti suhu, kelembapan, kadar polusi dan sebagainya. Selain itu lampu cerdas memiliki kemampuan otomatis untuk menyala ketika malam hari dan dapat meredup jika tidak ada pengguna jalan sehingga lampu dapat bekerja secara efisien dan hemat akan energi. Lampu cerdas atau *smart lighting* dapat di kontrol dan di monitoring dengan mudah, sehingga memudahkan dalam pemeliharaan. Contoh dari *smart lighting* diperlihatkan pada gambar 2.1



Gambar 2.1 *Smart Lighting*

2.2.1 Smart Lighting System

Sistem pada *Smart Lighting* terdiri dari beberapa proses pengiriman data. Proses pengiriman data ini melalui beberapa *access point* dengan menggunakan sistem komunikasi kooperatif. Sistem ini terdiri dari terminal pusat/induk dan beberapa terminal cabang. Terminal pusat ini dimaksudkan dengan terminal pada *smart lighting* yang memiliki seluruh data dari terminal-terminal lainnya serta dapat terhubung ke jaringan internet untuk mengirimkan data tersebut. Sedangkan terminal cabang merupakan terminal-terminal pada *smart lighting* yang tidak memiliki akses internet sehingga akan mengirimkan data yang dimilikinya ke tetangganya atau terminal di sebelahnya untuk menuju ke terminal pusat. Metode ini biasa disebut dengan *master node* dan *slave node*. Proses pengiriman data ini berawal dari terminal cabang terjauh dari terminal pusat yang mengirimkan data ke terminal cabang di sebelahnya, dan terminal cabang tersebut mengirimkan data lagi menuju terminal cabang di sebelahnya lagi, begitupun seterusnya hingga data menuju pada terminal pusat. Terminal pusat adalah terminal yang memiliki *gateway* untuk mengakses pusat *control and monitoring*. Terminal pusat akan mengirimkan data tersebut melalui jaringan internet menuju ke server monitoring. Metode ini biasa disebut dengan istilah *master node* dan *slave node*. Terminal Pusat merupakan *master node* dan terminal cabang merupakan *slave node*. Sistem dari *smart lighting* dari setiap terminal menuju pusat *monitoring* ditunjukkan pada gambar 2.2.

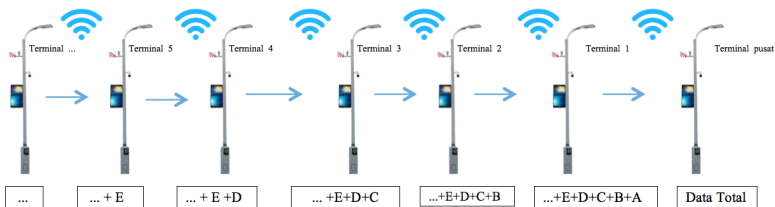


Gambar 2.2 Sistem *Smart Lighting*

2.3 Sistem Komunikasi Kooperatif

Sistem komunikasi kooperatif digunakan untuk mengurangi pengaruh fading pada kanal nirkabel, sehingga memperbaiki kinerja sistem. Sistem komunikasi kooperatif merupakan suatu metode yang memanfaatkan antenna dari pengguna lain dengan prinsip transmit diversity untuk mendapatkan suatu antenna virtual jamak dengan menggunakan lintasan yang berbeda dalam mengirimkan suatu data. Sehingga dibentuk suatu distribusi banyak antenna. Tujuannya adalah untuk mendapatkan informasi tambahan yang dapat membantu proses penjeremahan informasi di sisi penerima.

Pada sistem komunikasi kooperatif, pengirim mengirimkan informasi secara broadcast ke penerima dan ke relay yang merupakan pengguna lain dalam area yang berdekatan. Kemudian sinyal yang diterima oleh relay akan diolah terlebih dahulu diolah untuk kemudian dikirimkan ke penerima. Sinyal yang diterima dari pengirim maupun dari relay akan mengalami proses penggabungan di penerima. proses relaying yang dilakukan pada sistem komunikasi kooperatif dibagi menjadi dua macam yaitu *amplify and forward* dan *decode and forward*. komunikasi kooperatif pada *smart lighting* ditunjukkan pada gambar 2.3.



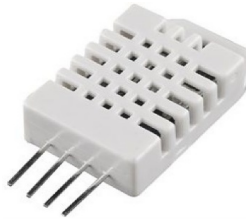
Gambar 2.3 Komunikasi Kooperatif pada Smart Lighting [6]

2.4 Sensor

Sensor adalah sesuatu yang digunakan untuk mendeteksi adanya perubahan lingkungan fisik atau kimia. Pada lampu cerdas atau *smart lighting* sensor berfungsi untuk memberikan informasi mengenai kondisi lingkungan di area tertentu sesuai dengan letak dari lampu cerdas tersebut. Berikut adalah sensor yang digunakan pada lampu cerdas atau *smart lighting*.

2.4.1 Sensor Suhu dan Kelembapan

Sensor temperatur dan kelembapan adalah sensor untuk mengetahui kondisi suhu dan kelembapan pada suatu area tertentu, salah satu dari sensor temperatur dan kelembapan adalah DHT22, dimana sensor digital ini dapat mengukur suhu dan kelembapan udara di sekitarnya. Sensor ini dapat digunakan pada Arduino dan memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam OTP program memory, sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan koefisien tersebut dalam kalkulasinya. Gambar 2.4 merupakan gambar dari modul DHT22.



Gambar 2.4 Sensor DHT22

DHT22 termasuk sensor yang memiliki kualitas baik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-interferensi. Ukurannya yang kecil membuat produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu dan kelembapan, adapun spesifikasi dari sensor DHT22 ditunjukkan oleh Tabel 2.4 berikut ini:

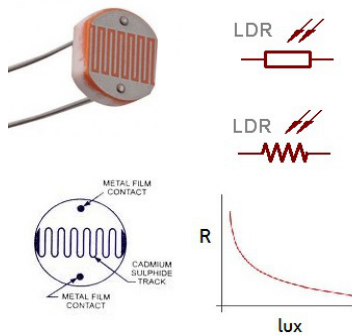
Input Voltage	3.3-6V
Measuring current	1-1.5mA
Standby current	40-50 uA
Humidity	0-100% RH
Temperature range	-40 - 80 degrees C
Accuracy	+2% RH

Tabel 2.4 Spesifikasi sensor DHT22

Pada *smart lighting* sensor suhu dan kelembapan berfungsi untuk mendeteksi kondisi mengenai cuaca pada lingkungan penerangan jalan umum tersebut. Informasi mengenai suhu berupa nilai dengan satuan derajat celcius sedangkan informasi mengenai kelembapan berupa nilai dengan satuan kondisi kelembapan. Informasi mengenai data dari sensor akan dikirimkan ke pusat *monitoring*.

2.4.2 Sensor Intensitas Cahaya

Sensor intensitas cahaya adalah sensor yang mampu mengetahui besarnya intensitas cahaya yang diterimanya. Salah satu dari jenis sensor ini adalah *Light Dependent Resistor (LDR)*. Komponen elektronik ini resistansinya akan menurun jika ada penambahan intensitas cahaya yang mengenainya. Bentuk fisik dan simbol rangkaian dari sensor LDR ditunjukkan oleh Gambar 2.5.



Gambar 2.5 Sensor LDR

Pada *smart lighting*, sensor ini berfungsi untuk mengatur inntensitas cahaya lampu agar dapat disesuaikan dengan intensitas cahaya pada lingkungan penerangan jalan. Jika intesitas cahaya pada lingkungan jalan rendah maka sensor ini akan mendeteksi sehingga lampu untuk penerangan jalan akan meningkatkan intesitas cahayanya. Sebaliknya jika tingkat intesitas cahaya pada lingkungan tinggi sensor ini akan mendeteksi kondisi tersebut dan akan merendhkan intesitas cahaya pada lampu. Informasi yang dihasilkan oleh sensor ini adalah tingkat pencahayaan untuk mengatur intensitas cahaya pada lampu.

2.4.3 Sensor *Infrared*

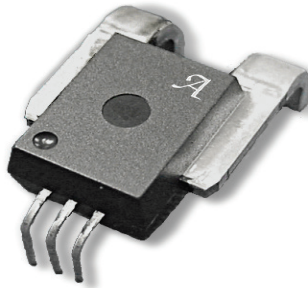
Sensor *Infrared* berfungsi sebagai sensor yang dapat mengetahui keberadaan suatu benda yang terdeteksi olehnya. Sensor ini terdiri dari 2 jenis yaitu sensor *infrared* aktif dan sensor *infrared* pasif. Sensor IR aktif berfungsi dengan mengirimkan energi dari *light emitting diode* (LED) ataupun diode laser. LED berfungsi untuk *non-imaging active IR detector* dan diode laser digunakan untuk *imaging active IR detector*. Detektor IR berfungsi atau mampu menghasilkan data perhitungan, ketersediaan, kecepatan dan kehadiran suatu benda pada siang ataupun malam hari. Sedangkan dioda laser berfungsi sebagai pembeda jenis kendaraan, seperti motor/mobil/ karena dapat memberikan data profil dan bentuk target. Kedua tipe detektor ini baik LED maupun diode laser menyinari targer da energi terpantul difokuskan ke detektor yang terdiri dari *pixel* atau *array pixel*. Data yang terukur kemudian diproses menggunakan algoritma *signal-processing* untuk mengetahui informasi yang diinginkan.

Pada sistem IR pasif, sistem tersebut mendeteksi energi yang dihasilkan oleh sebuah benda dan menggunakan algoritma *signal-processing* untuk memperoleh informasi yang diinginkan. Energi tidak dikeluarkan oleh IR pasif dalam proses deteksi. Yang mampu di deteksi oleh sistem IR pasif adalah data kehadiran, ketersediaan, dan perhitungan target/benda. Informasi yang dihasilkan dari sensor ini adalah nilai dari konsisi ada tidaknya kehadiran dari suatu benda.

Sensor *infrared* pada *smart lighting* berfungsi untuk mendeteksi pengendara atau pengguna jalan yang lewat, sehingga jika sensor tidak mendeteksi adanya pengguna jalan, maka lampu akan menurunkan intensitas cahaya untuk penghematan energi. Sebaliknya ketika lampu pada intensitas cahaya yang rendah dan terdeteksi adanya pengguna jalan maka lampu akan mengatur intensitas cahaya menjadi tinggi. Sehingga lampu menyala optimal hanya pada kondisi ketika dibutuhkan. Hal ini menyebabkan penghematan energi pada pencahayaan untuk penerangan jalan umum.

2.4.4 Sensor Arus Listrik

Sensor ini berfungsi untuk meningkatkan manajemen *fault* dan *maintenance* pada sistem *smart lighting*. Sensor ini digunakan untuk mengontrol kondisi lampu apakah lampu dalam keadaan menyala atau mati. Selain itu sensor ini berfungsi sebagai kontrol untuk mengetahui arus yang digunakan pada lampu agar nilai dari daya yang digunakan oleh lampu dapat dideketahui. Arus pada sistem bernilai 1.5 A, sehingga sensor ini cocok untuk mengetahui nilai arus yang mengalir pada sistem. Nilai dari pembacaan sensor untuk mengontrol lampu dibatasi antara nilai 1 dan 1.5 A. Informasi mengenai lokasi serta kondisi lampu dan status mengenai arus listrik akan dikirimkan menuju pusat monitoring agar jika terjadi kerusakan pada lampu, operator pada pusat *monitoring* dapat mengirimkan teknisi dengan mudah dan mengetahui letaknya secara jelas. Sensor yang digunakan untuk manajemen *fault* dan *maintenance* ini adalah sensor ACS756. Gambar dari sensor arus listrik ini ditunjukkan pada gambar 2.6. Informasi yang dihasilkan oleh sensor ini berupa identitas dari lampu, kondisi berfungsi atau tidaknya lampu, dan status dari arus listrik yang dihasilkan.



Gambar 2.6 Sensor Arus Listrik

Pada sistem *smart lighting* sensor ini berfungsi untuk mendeteksi energi yang dihasilkan oleh setiap waktu agar dapat terkontrol oleh pusat *monitoring*, hal ini dapat mempengaruhi pengurangan biaya pada *ontrol and maintenance* sistem pada smart lighting sehingga lebih efektif dan energi dan biaya pada setiap lampu dapat berkurang dengan adanya sensor pendeteksi arus listrik ini.

2.5 Protokol Komunikasi

Protokol Komunikasi yang dapat digunakan pada *Smart Lighting* bermacam-macam. Sistem yang disajikan dalam literatur ilmiah memungkinkan transmisi data untuk komunikasi nirkabel lokal menggunakan standar seperti Wi-Fi, ZigBee dan 6LoWPAN (IPv6 melalui Jaringan Area Pribadi Nirkabel), atau menggunakan jalur energi sebagai sarana komunikasi dengan menggunakan PLC (*Power Line Communications*). Namun PLC merupakan jaringan kabel sehingga jika salah satu jalur terputus maka informasi tidak dapat dikirimkan menuju *destination*. Dalam menerapkan pemantauan lampu jalan dan sistem kontrol, jarak antara tiang lampu jalan adalah sekitar 20-50 meter. protokol komunikasi apapun yang dapat mengirimkan data dari banyak sensor dan memastikan transfer data yang optimal pada jarak 20 hingga 50 meter atau lebih dapat dianggap sebagai kandidat yang baik untuk integrasi dalam suatu kontrol otomatis penerangan jalan. Tabel 2.4 akan menunjukkan perbedaan standar dan spesifikasi dari setiap protokol komunikasi.

Standard	ZigBee (802.15.4)	GPRS/GSM	Wi-Fi (802.11b)	Bluetooth (802.15.1)
Application Focus	Monitoring & Control	Wide Area Voice & Data	Web, Email, Video	Cable Replacement
System Resources	4KB-32KB	16MB+	1MB+	250KB+
Battery Life (days)	100-1.000+	1-7	5-5	1-7
Network Size	Unlimited	1	32	7
Bandwidth (Kb/s)	20-250	64-128+	11.000 +	720
Transmission Range	1-100+	1.000+	1-100	1-10+
Success Metrics	Reliability, Power, Cost	Reach, Quality	Speed, Flexibility	Cost, Convenience

Tabel 2.4 Perbandingan Standar Komunikasi

Wi-fi memiliki keunggulan dari segi jangkauan dan *transfer rate*. Dengan modul Wi-Fi, data dapat dikirim ke jangkauan yang lebih

luas dalam waktu yang singkat karena memiliki *transfer rate* yang dimiliki Wi-Fi bernilai 50 kali lebih besar daripada Bluetooth. Namun energi yang dibutuhkan juga sangat besar. Sedangkan pada Zigbee, *transfer rate* yang dimiliki hanya bernilai sampai 250 Kb. Namun *transfer rate* tersebut lebih dari cukup jika hanya digunakan untuk *sensor*. Sedangkan Energi yang dibutuhkan pun jauh lebih sedikit dibandingkan wifi.

2.5.1 Wi-Fi

Komunikasi dengan teknologi Wi-Fi digunakan standard komunikasi IEEE. Salah satu standard IEEE yang sering digunakan adalah keluarga IEEE 802.11. Standard komunikasi ini sudah digunakan sejak tahun 1985 oleh pemerintah Amerika Serikat. Teknologi jaringan ini merupakan yang paling umum digunakan. 802.11 adalah protkol spesifikasi jaringan area lokal nirkabel yang diresmikan oleh organisasi IEEE pada tahun 1997 dan digunakan untuk mendukung komunikasi nirkabel antara jaringan kabel dan jaringan nirkabel atau sesama jaringan nirkabel pada band ISM 2.4 GHz. Sejak peresmian dari spesifikasi ini, protkol ini telah mengalami dasar versi dari tipe a / b / g / n serta dilanjuti dengan peningkatan versi dalam aspek encoding / decoding dan modulasi / demodulasinya. Tingkat transmisi maksimum meningkat secara bertahap menjadi 11/54/600 Mbps yang berawal dari awal 1 ~ 2 Mbps. Pengembangan ini memiliki beberapa tambahan spesifikasi dalam aspek jaminan seperti kualitas layanan , jaringan roaming , keamanan jaringan, sumber daya nirkabel manajemen,interaksi antar jaringan, dan manajemen jaringan. Mempertimbangkan pasar yang sangat besar dari teknologi 802.11, penelitian dan pengembangan pada 802.11 adalah topik yang sering dibahas di kedua aspek akademisi maupun industri. Dalam penelitian jaringan nirkabel berdasarkan 802.11, simulasi kinerja jaringan sangat penting. Pekerjaan penelitian mengenai ini perlu memverifikasi kemungkinan itu dan kinerja strategi serta metode perutean akses saluran dalam model simulasi propagasi sinyal nirkabel, *fading* dan semacamnya agar sesuai dengan lingkungan dari sistem ketika di implementasikan.

2.5.2 Zigbee

ZigBee merupakan standar global baru untuk konektivitas nirkabel. Zigbee fokus pada standarisasi dan memungkinkan interoperabilitas produk. ZigBee adalah standar komunikasi yang menyediakan kemampuan jaringan jarak pendek yang hemat biaya.

Dikembangkan dengan memiliki penekanan pada hemat energi. ZigBee telah diperkenalkan oleh IEEE dengan standar IEEE 802.15.4 dan aliansi ZigBee memberikan standar umum ini agar dapat diaplikasikan. Contoh dari aliansi ZigBee dan semacamnya digunakan oleh perusahaan seperti Invensys, Honeywell, Mitsubishi Listrik, Motorola, dan Philips. ZigBee dibangun pada layer fisik (PHY) dan layer komunikasi medium access control (MAC) lapisan yang ditentukan oleh standar IEEE 802.15.4. Selain itu ZigBee dapat digunakan untuk topologi jaringan *mesh*, *star*, dan topologi *tree* dengan fitur keamanan data yang tinggi dan profil aplikasi yang mudah digunakan.

2.6 Estimasi Kapasitas Bandwidth

Kebutuhan atas bandwidth dari satu jaringan ke jaringan lainnya bisa bervariasi. Sangat penting menentukan berapa banyak bit per detik yang melintasi jaringan dan jumlah bandwidth yang digunakan tiap-tiap aplikasi agar jaringan bisa bekerja cepat dan fungsional. Bisa dibuktikan oleh banyak administrator jaringan, bandwidth untuk jaringan adalah salah satu factor penting dalam merancang dan memelihara LAN atau WAN yang baik. Tidak seperti server, yang bisa dikonfigurasi dan di konfigurasi-ulang sepanjang masa aktif suatu jaringan, bandwidth adalah salah satu dari elemen-elemen desain jaringan yang biasanya dioptimalkan dengan cara terbaik dengan mengkonfigurasi jaringan secara benar dari terminal luar. Bandwidth mengacu pada data rate yang didukung oleh koneksi jaringan yang terhubung ke jaringan. Biasanya diekspresikan dalam istilah bit per sekon (bps), atau terkadang pada byte per sekon (Bps).

Estimating Available Bandwidth (EAB) atau estimasi bandwidth yang tersedia merupakan hal yang penting pada suatu jaringan nirkabel. Karena EAB ini yang akan menentukan nilai *Quality of Service* (QoS) pada suatu jaringan. Sebelum menentukan koneksi baru pada jaringan wireless penting untuk diketahui batas maksimal pada kapasitas bandwidth yang tersedia pada jaringan yang menyediakan koneksi baru tersebut. Jika kapasitas maksimal pada bandwidth jaringan tersebut lebih besar dibandingkan dengan kapasitas bandwidth jaringan baru maka koneksi akan diterima, sebaliknya jika tidak maka tidak bisa. salah satu cara untuk menentukan kapasitas bandwidth pada suatu jaringan adalah dengan menghitung nilai rata-rata dari aplikasi yang digunakan.

2.7 Analisa Propagasi

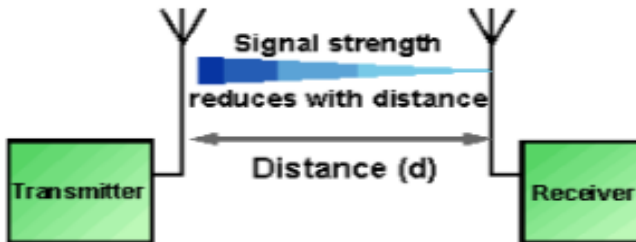
Dalam membangun sebuah jaringan komunikasi yang ideal terdapat beberapa faktor yang patut diperhitungkan. Jaringan komunikasi yang baik tentunya memiliki sistem transmisi jaringan yang ideal. Kualitas sistem transmisi yang ideal dapat dipengaruhi oleh beberapa hal, seperti jarak transmisi, jenis area transmisi, banyaknya distraksi dan interferensi dan sebagainya. Untuk menentukan kualitas transmisi dalam sebuah jaringan dapat diukur menggunakan analisa propagasi dan budget link.

Propagasi adalah transmisi atau penyebaran sinyal dari suatu tempat ke tempat lain dengan menggunakan media perambatan atau biasa disebut saluran transmisi. Saluran transmisi dapat berupa fisik maupun non fisik. Saluran transmisi berupa fisik yaitu seperti kawat konduktor, kabel koaksial dan sebagainya. Sedangkan untuk media transmisi non fisik yaitu sinar dan gelombang radio.

Propagasi gelombang radio merupakan proses perambatan gelombang radio dari pemancar ke penerima. Transmisi sinyal dengan media non fisik memerlukan antenna untuk meradiasikan sinyal radio tersebut ke udara bebas dalam bentuk gelombang elektromagnetik. Gelombang elektromagnetik akan merambat melalui udara bebas menuju antenna penerima dengan mengalami peredaman selama berada pada lintasannya, sehingga ketika sampai di antenna penerima, energi sinyal sudah melemah.

2.7.1 Free Space Path Loss

Pathloss adalah suatu metode yang digunakan untuk mengukur suatu loss sinyal yang disebabkan oleh cuaca, kontur tanah, kondisi medan transmisi dan lain-lain, agar tidak mengganggu pemancaran antar 2 buah antenna yang saling berhubungan. Nilai pathloss menunjukkan level sinyal yang melemah (mengalami attenuation) yang disebabkan oleh propagasi free space seperti refleksi, difraksi, dan scattering. Path loss sangat penting dalam perhitungan Link Budget, ukuran cell, ataupun perencanaan frekuensi. Faktor-faktor yang mempengaruhi nilai level daya dan pathloss adalah jarak pengukuran antara Tx dan Rx, tinggi antena (Tx dan Rx), serta jenis area pengukuran. Gambar 2.7 akan menunjukkan ilustrasi mengenai *free space path loss* pada media transmisi antara *transmitter* dan *receiver*.



Gambar 2.7 Ilustrasi Free Space Path Loss

Path loss digunakan untuk melakukan planning dalam membuat sebuah jaringan baik yang bersifat *point to point* (titik ke titik) ataupun *multipoint* (banyak titik). Pathloss adalah pengurangan kepadatan daya (atenuasi) dari sebuah gelombang elektromagnetik karena penyebarannya dalam ruangan. Ilustrasinya adalah seperti ketika batu dijatuhkan ke dalam air, dimana disekitar titik batu tersebut jatuh akan muncul gelombang-gelombang yang bergerak menjauhi titik batu jatuh yang semakin lama menjadi semakin lemah hingga gelombang tersebut tidak terlihat lagi. Begitu juga pada transmisi sinyal, semakin jauh jarak transmisi maka semakin lemah gelombang sinyal yang diterima.

Free space path loss adalah hilangnya kuat sinyal elektromagnetik yang berada dalam satu garis lurus tanpa hambatan diantaranya. Dalam standard IEEE Std 145-1983 "*standard definitions of term for antennas*" didefinisikan bahwa FSPL adalah hilangnya tenaga antara dua radiator isotropic dalam ruang bebas yang diekspresikan dalam perbandingan daya. FSPL dinyatakan dalam satuan dB.

2.7.2 Propagasi Two-Ray Ground

Proses pengiriman data yang dikirimkan oleh antenna pengirim menuju antenna penerima tidak selalu berjalan dengan lancar, terutama jika di lingkungan antara penerima dan pengirim terdapat banyak penghalang. Model propagasi *two-ray ground* mempertimbangkan hal tersebut dengan menggunakan 2 faktor. Faktor tersebut yaitu dengan pertimbangan menggunakan jalur secara langsung (*line-of-sight*) dan jalur pantulan tanah.

2.7.3 Propagasi Shadowing

Propagasi *shadowing* merupakan model propagasi yang dimana menggambarkan kondisi lingkungan sesungguhnya. Model propagasi ini menyediakan parameter kondisi lingkungan yang lebih spesifik daripada model propagasi lainnya. Hal ini disebabkan model propagasi *shadowing* terdapat nilai dari parameter *path loss exponent* (β) dan *shadowing deviation* (σ_{dB}). Parameter ini bernilai berbeda-beda sesuai dengan kondisi dari lingkungan, seperti kondisi *free space*, *urban area*, *line-of-sight*, *obstructed* serta kondisi di dalam atau luar dari bangunan. Nilai dari parameter *path loss exponent* (ρ) dan *shadowing deviation* (σ_{dB}) ditunjukkan pada tabel 2.5.

Lingkungan		ρ	σ (dB)
Outdoor	<i>Free Space</i>	2	4 – 12
	<i>Urban</i>	2.7 – 5	
Indoor	<i>Line-of-Sight</i>	1.6 – 1.8	3 – 6
	<i>Obstruted</i>	4 – 6	6.8

Tabel 2.5 Parameter *Path Loss Exponent* dan *Shadowing Deviation*

2.8 Delay (Latency)

Delay (Latency) adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media transmisi (kabel koaksial, fiber optik, gelombang radio), kongesti atau juga waktu proses yang lama yang disebabkan oleh komponen jaringan seperti router, modem, hub, dan lainnya. Jenis-jenis Delay:

- Algorithmic Delay: Delay yang disebabkan oleh standar codec yang digunakan.
- Packetization Delay: Delay yang disebabkan oleh pengakumulasian bit sample ke frame.
- Serialization Delay: Delay yang disebabkan karena adanya waktu yang dibutuhkan untuk penstransmisian paket IP dari sisi pengirim.
- Propagation Delay: Delay ini terjadi karena perambatan atau perjalanan. Paket IP di media transmisi menuju alamat tujuan.
- Coder (Processing) Delay: Waktu yang diperlukan oleh Digital Signal Processing (DSP) untuk mengompres sebuah block PCM, Nilainya bervariasi bergantung dari codec dan kecepatan prosessor.

2.9 Throughput

Throughput merupakan kecepatan (rate) transfer data efektif yang diukur dalam bps. Throughput merupakan jumlah total kedatangan paket yang sukses, yang di amati pada penerima selama interval waktu tertentu dibagi oleh interval waktu tersebut. Jika *latency* mengukur waktu yang dibutuhkan dari mulai pengiriman sampai selesai, throughput adalah waktu keseluruhan dari seluruh proses dalam waktu tertentu.

$$\text{Throughput} = \frac{\text{Jumlah Paket yang dikirim}}{\text{Waktu Pengiriman Paket}} \quad [2.1]$$

Throughput dan *Latency* saling berhubungan satu sama lain. Karena pada pengiriman paket dengan jumlah yang besar dapat mengakibatkan nilai *throughput* yang besar dibandingkan dengan pengiriman data dengan paket berukuran kecil karena *latency* bernilai besar. Jika data dikirim secara kontinyu, nilai *latency* akan cenderung kecil terhadap nilai *throughput*. Akan tetapi jika pengiriman menunggu *acknowledgment* (ack) sebelum paket selanjutnya akan dikirim, maka dapat menyebabkan tingginya nilai *latency* berdampak menurunkan nilai *throughput*.

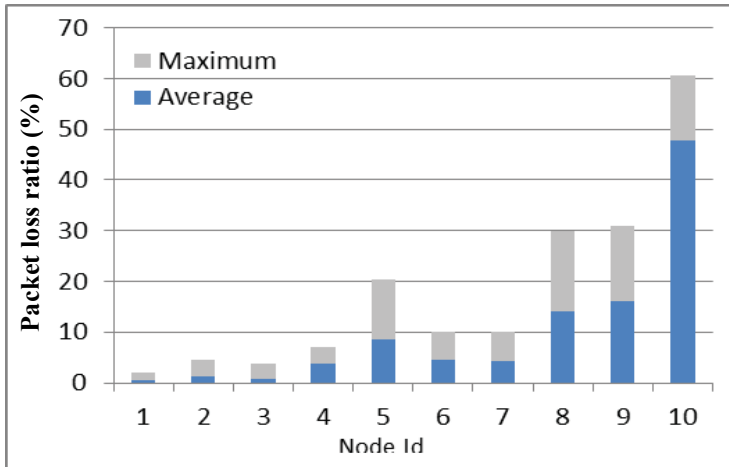
2.10 Packet Loss Ratio

Rasio paket yang hilang atau biasa disebut dengan *packet loss ratio* merupakan perbandingan jumlah paket yang hilang ketika proses pengiriman paket dengan total paket yang dikirimkan dalam suatu periode waktu tertentu. Dapat juga dihitung dengan cara mengurangi jumlah paket keseluruhan yang dikirim dengan paket jumlah paket yang diterima pada *node* tujuan. Nilai dari Rasio tersebut dipengaruhi oleh beberapa faktor seperti ukuran paket, jarak pengiriman dan mobilitas dari *node*.

$$\text{Packet Loss Ratio (\%)} = \frac{\text{Jumlah paket hilang}}{\text{Jumlah Paket dikirim}} \times 100\% \quad [2.2]$$

Jika Paket terkirim sempurna dan tidak adanya nilai dari paket yang hilang maka dapat dikatakan bahwa *node* penerima menerima seluruh paket sebelum waktu periode pengiriman habis. Jika paket tidak terkirim sempurna maka penerima belum menerima seluruh paket yang

dikirim dan periode telah habis sehingga menyebabkan adanya paket yang hilang. Nilai *packet loss ratio* pada sistem komunikasi kooperatif dapat dilihat pada gambar 2.8.



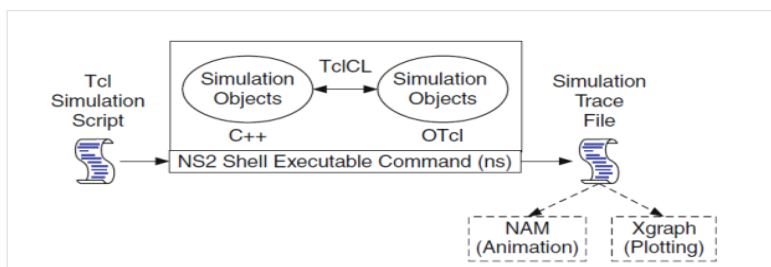
Gambar 2.8 Nilai *Packet Loss Ratio* (%)

2.11 Network Simulator-2 (NS-2)

Network simulator adalah alat simulasi berbentuk *software* yang memiliki kejadian diskrit dan berfungsi untuk meneliti mengenai jaringan. Jaringan yang dapat disimulasikan dapat berupa jaringan nirkabel (*wireless*) ataupun jaringan kabel (*wired*). NS-2 menyediakan berbagai fungsi skema *routing*, protokol *multicast*, dan protokol IP seperti UDP, TCP, RTP, dan SRM melalui kabel maupun nirkabel serta satelit. NS-2 juga menyediakan beberapa algoritma seperti *routing* dan *queuing*. *Network Simulator* dapat dijalankan pada sistem operasi seperti Linux, Windows, dan MacOs.

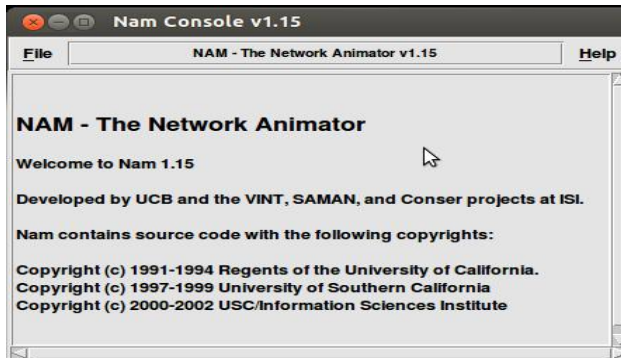
Network Simulator (NS) pertama kali dikembangkan di University of California Berkeley (UCB) serta mendapat dukungannya oleh Defense Advanced Research Project Agency (DARPA) Amerika Serikat. Setelah itu network simulator berkembang dan dijadikan alat simulasi untuk jaringan oleh banyak peneliti di dunia. Seiring dengan perkembangan zaman NS semakin berkembang menjadi NS-2 hingga NS-3.

Pada Network Simulator 2 terdapat beberapa file program yang bekerja pada simulasi jaringan ini seperti Tool command Language (Tcl), Tool kit (Tk), Object Tool Command Language (Otc), Tool Command Language / C++ Interface (Tclcl), Network Animator (NAM). NS-2 Menggunakan bahasan TCL dan C++ dalam skrip pemrogramannya, TCL digunakan untuk membuat program berdasarkan objeknya, pada umumnya bahasa ini digunakan untuk membuat struktur dan topologi jaringan. Bahasa TCL dapat dikatakan lebih mudah digunakan dibandingkan C++ untuk menentukan konfigurasi parameter yang dibutuhkan, namun bahasa TCL tidak memiliki arsitektur protokol sehingga kurang cocok sebagai bahasa tunggal yang digunakan dalam penelitian. Sehingga C++ adalah bahasa pemrograman yang paling banyak digunakan pada NS-2, apabila ada protokol baru yang ingin di simulasikan maka protokol baru tersebut dapat dibuat dengan bahasa C++ kemudian meletakkannya pada perpustakaan NS dan mendaftarkan protokol baru tersebut ke perpustakaan NS (*make file*). Pada gambar 2.9 akan diperlihatkan arsitektur NS-2.



Gambar 2.9 Arsitektur NS-2

Keluaran dari simulasi NS yaitu berupa Network Animator (NAM) dan File trace (.tr). File keluaran tersebut memiliki fungsi masing-masing yang berbeda. Network animator berfungsi sebagai grafis simulasi sedangkan file trace berfungsi sebagai analisa numerik yang digunakan untuk menganalisis kinerja jaringan. Namun file trace hanyalah file mentah yang berbasis tulisan sehingga diperlukan perangkat lunak untuk menganalisis hasilnya. Contoh perangkat lunak yang dapat digunakan adalah Tracegraph, xgraph ataupun yang berbasis online seperti jgyan *analyzer*. Tampilan dari keluaran simulasi berupa NAM akan diperlihatkan pada gambar 2.10.



Gambar 2.10 Tampilan NAM pada NS-2

2.11.1 Protokol dan Model Pada NS-2

Beberapa model yang dapat dibuat dan dijalankan pada NS-2 adalah sebagai berikut:

- Jaringan kabel dan Nirkabel
- Mobile Network Ad-Hoc Routing
- Proses Antrian seperti Drop-Tail, RED
- Quality of Service (QoS)
- Permodelan Trafik

2.11.2 Transport Agent pada NS-2

Pada jaringan internet terdapat empat lapisan komunikasi TCP/IP, yaitu: lapisan aplikasi, lapisan transport, lapisan IP dan jaringan. Lapisan Transport merupakan lapisan komunikasi yang mengatur komunikasi data yang digunakan oleh lapisan aplikasi di atasnya. NS mensimulasikan lapisan transport dengan objek simulasi bernama transport agent . Pada simulasi pengiriman data, transport agent tidak dapat berdiri sendiri sehingga membutuhkan lapisan aplikasi di atasnya yang berfungsi sebagai pembangkit trafik. Adapun protokol lapisan transport data yang didukung NS-2 sebagai berikut:

1. TCP (Transport Control Protocol)

Transport Control Protocol merupakan protokol transport yang andal karena mempunyai mekanisme yang dapat memastikan paket dapat diterima oleh client . Pada saat TCP sedang dikirimkan ke node penerima, protokol ini akan memberikan state acknowledgement (ACK).

Jika ACK ini tidak diterima oleh node pengirim, maka secara otomatis ACK ini akan dikirim ulang dalam selang waktu tertentu.

2. UDP (User Datagram Protocol)

User Datagram Protocol adalah lapisan transport yang tidak andal, connectionless, dan merupakan kebalikan dari lapisan transport TCP. Setiap aplikasi socket dapat mengirimkan paket – paket berupa datagram. Istilah datagram diperuntukkan terhadap paket dengan koneksi yang tidak andal. Koneksi tidak andal adalah koneksi yang tidak akan mengirimkan keterangan saat pengiriman data gagal.

UDP tidak menjamin kevalidan data saat data sampai ke node penerima. Datagram yang sampai mempunyai kemungkinan tidak sampai, rusak, duplikasi atau hilang tanpa diketahui penyebabnya. Untuk itu penggunaan UDP lebih tepat saat data – data yang dikirim merupakan data kecil dalam jumlah banyak. Dengan itu, maka data akan sampai lebih cepat dan lebih efisien mengingat sifat UDP yang tidak akan melakukan pengecekan terhadap paket tersebut.

2.11.3 Level Aplikasi pada NS-2

Terdapat dua tipe dasar aplikasi pada NS-2, diantaranya:

1. Simulated Application

Simulated Application adalah aplikasi yang dapat langsung disimulasikan pada NS-2, yaitu FTP dan Telnet. FTP dibangun untuk mensimulasikan *bulk data transfer*. Sedangkan Telnet adalah aplikasi yang diatur oleh *transport agent* dengan jumlah paket yang ditransmisikan dan diatur oleh mekanisme *flow control* dan *congestion control* TCP.

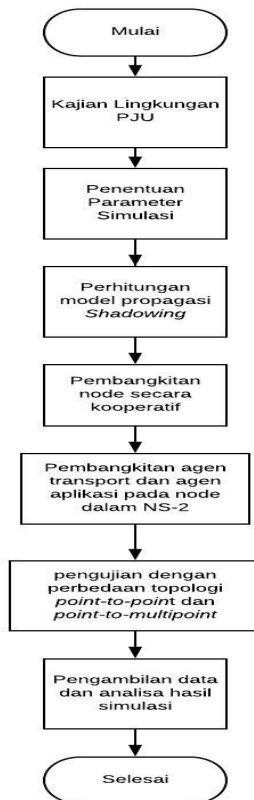
2. Generator Traffic

Pembangkitan trafik pada NS-2 terdiri atas empat tipe yang diantaranya eksponensial, *pareto*, CBR dan *Traffic Trace*. Eksponensial adalah trafik yang dibangkitkan pada waktu kedatangan antar paket sesuai dengan fungsi eksponensial. Sedangkan *pareto* adalah trafik yang dibangkitkan pada waktu kedatangan antar paket sesuai dengan fungsi yang dimiliki oleh *pareto*. Selain kedua fungsi tersebut, terdapat juga trafik CBR yaitu trafik yang dibangkitkan secara kontinyu dengan *bit rate* konstan dan *traffic trace* yang merupakan trafik dari sebuah *file trace*.

BAB 3 PERANCANGAN DAN SIMULASI SISTEM

3.1 Diagram Alir Perancangan

Pada bab ini membahas mengenai perancangan simulasi sistem *Smart Lighting* dengan bantuan *Network Simulator-2* (NS-2). Kemudian penentuan parameter dibuat agar dapat mewakili kondisi lingkungan penerangan jalan yang sesungguhnya.



Gambar 3.1 Diagram Alir Perancangan Simulasi

Gambar 3.1 merupakan diagram alir perancangan pada penelitian tugas akhir. Perancangan dimulai dengan menganalisa kajian lingkungan PJU, setelah itu membuat desain simulasi beserta parameternya agar sesuai dengan kondisi pada PJU. Setelah itu dijalankan simulasi menggunakan perangkat lunak Network Simulator dengan topologi yang berbeda-beda. Setelah berhasil, analisis dilakukan menggunakan bantuan perangkat lunak tracegraph agar data dapat di analisa sehingga didapatkan kesimpulan. Parameter yang dimaksud untuk melakukan simulasi adalah jarak antar lampu, jumlah lampu, dan *routing* protokol yang digunakan. Setelah itu, membuat simulasi berdasarkan parameter tersebut untuk sistem kooperatif pada *smart lighting*.

Pengujian diawali dengan jumlah *node* terendah yaitu 3 *node* hingga sekian *node* hingga hasil dari simulasi memiliki nilai *packet loss ratio* atau *end-to-end delay* masih berada pada standar yang telah ditetapkan. Jika hasil dari simulasi pada jumlah *node* dibawah 9 bernilai melebihi nilai *packet loss ratio* atau *end-to-end delay* yang telah ditetapkan, maka simulasi dibatasi hingga 9 *node*.

3.2 Kajian Lingkungan Penerangan Jalan Umum

Lingkungan penerangan jalan umum terdapat dua jenis menurut lokasinya yakni penerangan jalan umum untuk pejalan kaki, dan penerangan jalan umum untuk kendaraan. Fokus kajian lingkungan penerangan jalan umum pada tugas akhir ini adalah pada jenis penerangan jalan umum untuk kendaraan di jalan raya. Topologi Penerangan Jalan Umum menyesuaikan dengan kondisi pada jalan ataupun lingkungan yang membutuhkan pencahayaan, topologi tersebut akan berbeda sesuai dengan kondisi dan letak dari penerangan jalan umum itu sendiri. Pada tugas akhir ini, topologi penerangan jalan umum yang digunakan adalah penerangan jalan umum untuk daerah perkotaan khususnya penerangan jalan untuk kendaraan. Node pusat akan menjadi node untuk menerima seluruh data dari node terminal, dan masing-masing node terminal akan mengirimkan data secara kooperatif menuju node pusat. Nilai dari jarak antar lampu untuk penerangan jalan umum ditentukan berdasarkan Standar Nasional Indonesia, yaitu memiliki nilai yang berbeda-beda tergantung jenis dari lampu yang

digunakan. Nilai tersebut terdiri dari nilai 30 meter, 40 meter dan 50 meter. Sehingga pada simulasi ini dilakukan pengujian komunikasi kooperatif dengan jarak antar *node* 30 meter, 40 meter, dan 50 meter.

3.3 Penentuan Parameter Simulasi

Penentuan parameter ini akan mempengaruhi kinerja dari sistem smart lighting. Penentuan parameter ini berdasarkan parameter pada jurnal dan standar IEEE. Pada penulisan program tcl, program yang pertama ditulis adalah penulisan paramter simulasi dan kemudian dilanjutkan dengan pembangkitan node-node yang akan digunakan. Paramater beserta infonya dapat dilihat pada tabel 3.1

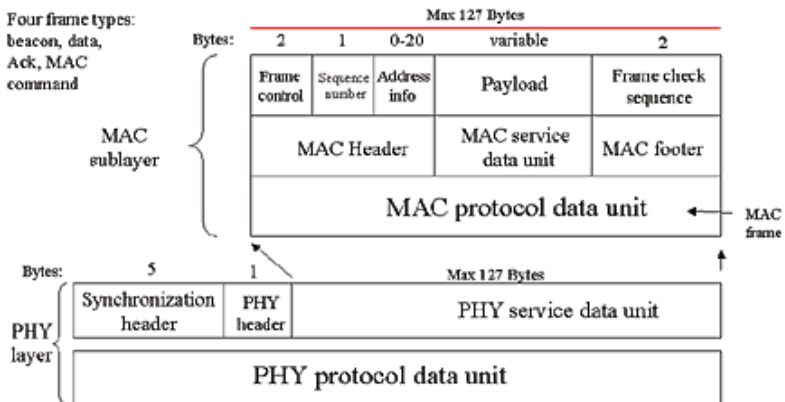
Parameter	Keterangan
<i>Channel type</i>	Wireless Channel
<i>Propagation model</i>	<i>Shadowing</i>
<i>Phy type</i>	Network/WirelessPhy
<i>Mac protocol type</i>	IEEE 802.15.4
<i>Queue type</i>	DropTail
<i>Link layer type</i>	LL
<i>Antenna type</i>	OmniAntenna
<i>Max packet ini queue</i>	50
<i>Routing Protocol</i>	AODV
<i>Number of Node</i>	3, 5, 7, 9
<i>Simulation time</i>	10 s

Tabel 3.1 Parameter Simulasi pada NS-2

Penjelasan mengenai parameter simulasi diatas adalah sebagai berikut:

1. Pemilihan Parameter kanal yaitu *wireless* karena sistem dari smart lighting pada tugas akhir ini menggunakan komunikasi kooperatif dimana sistem ini bekerja menggunakan jaringan nirkabel.
2. Simulasi dilakukan dengan menggunakan lapisan fisik standrard IEEE MAC Layer 802.15.4. Pemilihan parameter tersebut disesuaikan dengan standar untuk sistem komunikasi nirkabel pada sistem smart lighting.

3. Tipe antrian yang digunakan pada simulasi ini yaitu tipe DropTail. Tipe Antrian ini bekerja dengan cara melepas jumlah data yang melebihi kapasitas antrian, sehingga paket akan di *dropped*.
4. Tipe *Routing* yang digunakan dari simulasi ini yaitu Ad hoc On-Demand Distance Vector (AODV). Performa AODV pada standar IEEE 802.15.4 lebih optimal dibandingkan protokol lainnya yaitu DSDV maupun DSR. [11]
5. Jumlah Node dalam simulasi ini bermacam-macam. Mulai dari node yang sedikit hingga ke jumlah node yang berjumlah lebih banyak. Simulasi ini dimulai dari jumlah node 3,5,7,9.



Gambar 3.2 Frame struktur dalam satu paket data

Penentuan besarnya nilai paket yang dikirim oleh *sensor node* bergantung pada *packet structure frame* yang digunakan oleh media transmisi. Sesuai dengan gambar 3.2 jumlah paket data pada dalam satu paket *frame* adalah 5 byte untuk *Synchroniztion header*, 1 byte untuk *PHY header* pada lapisan fisik. Nilai *PHY Service Data Unit* terdiri dari beberapa frame pada lapisan *MAC*. Nilai tersebut terdiri dari 2 byte untuk *frame control*, 1 byte untuk *sequence number*, 0-20 byte untuk *Adreess info*, pada simulasi ini nilai tersebut diberikan nilai 8 byte untuk informasi mengenai alamat dari sumber dan tujuan, *frame* selanjutnya adalah 2 byte untuk *Frame Check Sequence*. Besarnya nilai Payload

berdasarkan hasil pembacaan dari setiap sensor yang berjumlah 4 sensor. Proses Pengiriman paket pada standar 802.15.4 adalah dari bentuk ASCII ke biner. Sehingga nilai dari pembacaan sensor dihitung berdasarkan jumlah digit yang dihasilkan yaitu bernilai 3 digit pada sensor suhu, 4 digit untuk sensor kelembapan, 5 digit untuk sensor arus, 2 digit untuk sensor *infrared*, dan 2 digit untuk sensor cahaya, setiap digit bernilai 8 bit atau setara dengan 1 *Byte*. Sehingga nilai dari *payload* adalah 16 *Byte*. Maka dapat ditentukan nilai satu paket *frame* adalah 34 *Byte*.

3.3.1 Perhitungan Model Propagasi dan Link Budget

Parameter selanjutnya yang perlu ditentukan selain parameter pada sub bab sebelumnya adalah perhitungan untuk model propagasi pada sistem smart lighting. Model propagasi yang digunakan pada simulasi di network simulator adalah untuk memprediksi kuat daya yang diterima. Pada layer fisik dari setiap node terdapat *threshold* penerimaan paket. Sehingga ketika paket diterima dan kuat daya tersebut bernilai lebih rendah daripada *threshold*, maka pengiriman paket tersebut akan gagal dan akan dilepas (*dropped*) oleh MAC layer. Pada NS-2 terdapat tiga model propagasi antara lain *free space*, *two-ray ground*, dan *shadowing*. Pada simulasi ini digunakan model propagasi yaitu propagasi *shadowing*. Model propagasi *shadowing* digunakan karena untuk mensimulasikan kondisi lingkungan sesungguhnya yaitu pada lingkungan *outdoor* urban. perhitungan jarak maksimal tidak dapat ditentukan secara langsung, namun harus menggunakan perhitungan berdasarkan batas daya yang dapat diterima oleh penerima. Nilai daya yang diterima pada suatu jarak tertentu dapat dihitung menggunakan rumus berikut.

$$\frac{Pr(d)}{Pt} = -10\rho \log 10 \frac{d}{d0} + X \quad [3.1]$$

Keterangan :

- Pr = *Power Receiver* (dB)
- d = Jarak antar node (km)
- Pt = *Power Transmitter* (dB)

- ρ = *Path Loss Exponent*
 X = *Gaussian Random Variable with zero mean*

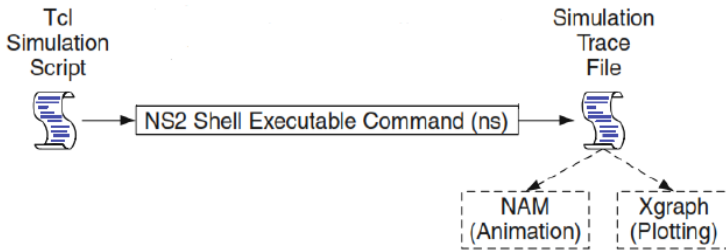
Setelah nilai dari link budget ditentukan maka diperlukan parameter lain untuk input pada model propagasi untuk menjalankan simulasi pada Network Simulator-2. Penentuan Parameter lainnya didapatkan dari standar dari IEEE 802.15.4. Sehingga didapatkan parameter model propagasi yang dapat dijalankan pada NS-2 sesuai pada tabel 3.2.

Parameter	Nilai
Transmit Antenna Gain	1
Receive Antenna Gain	1
System Loss Factor	1.0
Frequency Channel	2.45 GHz
Bandwidth Data Rate	250 Kb
Transmit Power	60 mW
Receive Power Threshold	2.4164e-11
Data Frames Rate	100 Kb
Control Frames Rate	50 Kb

Tabel 3.2 Parameter Propagasi pada NS-2

3.4 Desain Simulasi pada NS-2

Perancangan desain simulasi pada tugas akhir ini menggunakan perangkat lunak NS-2. Sebelum menjalankan simulasi, ada beberapa hal yang harus dilakukan yaitu memahami cara kerja dari NS-2 terlebih dahulu setelah itu mendesain simulasi yang akan dijalankan. Pengerjaan perancangan simulasi ini dimulai dengan membuat model sistem komunikasi kooperatif dengan membuat dan mendefinisikan parameter sistem lalu membangkitkan node yang terdiri dari *node* pusat dan *node* terminal. Node terminal akan dibangkitkan jumlahnya sesuai dengan tabel 3.1. Setiap topologi akan di analisa kinerja jaringannya, kinerja yang akan dianalisa adalah nilai *packet sent delay*, *packet loss*, *packet drop* dan *end-to-end* delay. Langkah-langkah ini dibuat dengan menggunakan bahasa C++ yang disimpan dalam bentuk file Tcl agar dapat dijalankan oleh perangkat lunak NS-2. Gambar 3.2 menjelaskan secara singkat model sistem simulasi dari NS-2.



Gambar 3.3 Model Sistem Menggunakan NS-2

Hasil dari simulasi ini akan menunjukkan dua tipe *file*. Yaitu NAM dan tr. Nam akan memperlihatkan peristiwa – peristiwa yang terjadi berupa animasi dari sistem yang sudah dirancang dan tr akan menunjukkan hasil berupa data atau informasi mengenai simulasi.

3.4.1 Pembangkitan Node

Setelah semua parameter telah ditentukan maka masuk ke tahap pembangkitan node. Pada pembangkitan node, node dapat diatur sesuai dengan kebutuhan, seperti jumlah node, pergerakan node, letak node, ukuran node dan sebagainya. Pada simulasi ini parameter untuk pembangkitan node hanyalah mengatur jumlah node, letak node dan ukuran node untuk membedakan node terminal dan node pusat. Untuk pergerakan node di atur agar node tidak bergerak karena pada simulasi ini tidak membutuhkan node yang bergerak. Pada gambar 3.3 akan ditampilkan kode untuk mengatur parameter node tersebut.

Langkah pertama yang dilakukan untuk membuat node adalah dengan mengetik format “set nama_node [\$ns node]”. Pada contoh di gambar 3.3 nama node akan di atur sebagai “n0” dan “n1”. Kemudian langkah kedua adalah mengatur lokasi node. Pada simulasi ini terdiri dari 2 lokasi pada penentuan node. Yaitu lokasi pada sumbu X dan sumbu Y karena topologi dari simulasi ini hanyalah dengan dua dimensi. Pengaturan lokasi tersebut dengan format “\$nama_node set X_koordinat pada sumbu X” untuk lokasi pada sumbu X. Untuk lokasi pada sumbu Y diatur dengan format “\$nama_node set Y_koordinat pada sumbu Y”.

Lokasi Sumbu Z diberi nilai 0 karena pada simulasi ini hanya menggunakan dua dimensi.

```
set n0 [$ns node]
$n0 set X_ 203
$n0 set Y_ 397
$n0 set Z_ 0.0
$ns initial_node_pos $n0 10
set n1 [$ns node]
$n1 set X_ 403
$n1 set Y_ 397
$n1 set Z_ 0.0
$ns initial_node_pos $n1 10
set n2 [$ns node]
$n2 set X_ 603
$n2 set Y_ 397
$n2 set Z_ 0.0
$ns initial_node_pos $n2 10
set n3 [$ns node]
$n3 set X_ 803
$n3 set Y_ 397
$n3 set Z_ 0.0
```

Gambar 3.4 Kode Pembangkitan *Node*

Untuk menentukan ukuran dari node hanya diatur dengan istilah `initial node pos` dengan cara menuliskan format “`$ns initial_node_pos $nnode ukuran node`”. Pada contoh nilai dari ukuran node diatur dengan nilai sebesar 3. Setelah membangkitkan node dengan nilai yang diinginkan diperlukan untuk mengatur nilai parameter dari node tersebut seperti pada gambar 3.4. Format dari parameter tersebut yang perlu dituliskan adalah “`$ns node-config -jenis_parameter $nilai_parameter \`”.


```

#=====
#      Simulation parameters setup
#=====
Antenna/OmniAntenna set Gt_ 1 ;#Transmit antenna gain
Antenna/OmniAntenna set Gr_ 1 ;#Receive antenna gain
Phy/WirelessPhy set pathlossExp_ 2
Phy/WirelessPhy set std_db_ 12
Phy/WirelessPhy set L_ 1.0 ;#System Loss Factor
Phy/WirelessPhy set freq_ 2.452e9 ;#channel
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate
Phy/WirelessPhy set Pt_ 0.06 ;#Transmit Power
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power Threshold
Mac/802_15_4 set dataRate_ 0.1Mb ;#Rate for Data Frames
Mac/802_15_4 set basicRate_ 0.05Mb ;#Rate for Control Frames

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) 50 ;# max packet in ifq
set val(nn) 17 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 2220 ;# X dimension of topography
set val(y) 100 ;# Y dimension of topography
set val(stop) 10.0 ;# time of simulation end

```

Gambar 3.5 Kode Pengaturan Parameter *Node*

Nilai tersebut sesuai dengan pada parameter simulasi diawal. Tujuannya adalah untuk menghubungkan parameter dari simulasi tersebut dengan node yang sudah dibangkitkan agar node dapat bekerja sesuai dengan parameter yang diinginkan. Nilai dari parameter ini disesuaikan dengan standar IEEE 802.15.4 dan perhitungan propagasi yang telah ditentukan.

3.4.2 Koneksi Lapisan Transport dan Aplikasi

Setelah melakukan pembangkitan node dan mengatur parameter jaringan yang akan dibuat langkah selanjutnya adalah memberikan perintah pada masing-masing node sesuai dengan skenario simulasi. Node-node tersebut harus saling terhubung secara fisik satu sama lain agar komunikasi atau pengiriman paket dapat dilaksanakan.

Dalam menghubungkan antara sesama node diperlukan koneksi penghubung dari dua lapisan, yaitu lapisan transport dan aplikasi. Langkah pertama adalah menghubungkan node pada lapisan *transport*.

Dalam pembuatan koneksi di lapisan *transport* dengan menuliskan program pada format file Tcl. Cara menghubungkan antar node dilakukan secara manual. Node sumber dituliskan sebagai node yang bertindak dituliskan perintah kode TCP dan node tujuan diberikan perintah kode *Sink*. Pada Gambar 3.6 akan diberikan contoh dari kode untuk mengatur node dengan agen transport.

```
#=====
#           Agents Definition
#=====
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink4 [new Agent/TCPSink]
$ns attach-agent $n1 $sink4
$ns connect $tcp0 $sink4
$tcp0 set packetSize_ 32

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink5 [new Agent/TCPSink]
$ns attach-agent $n2 $sink5
$ns connect $tcp1 $sink5
$tcp1 set packetSize_ 48

#Setup a TCP connection
set tcp2 [new Agent/TCP]
$ns attach-agent $n2 $tcp2
set sink6 [new Agent/TCPSink]
$ns attach-agent $n3 $sink6
$ns connect $tcp2 $sink6
$tcp2 set packetSize_ 60

#Setup a TCP connection
set tcp3 [new Agent/TCP]
$ns attach-agent $n3 $tcp3
set sink7 [new Agent/TCPSink]
$ns attach-agent $n4 $sink7
$ns connect $tcp3 $sink7
$tcp3 set packetSize_ 78
```

Gambar 3.6 Kode Pengaturan Agen *Transport*

Langkah selanjutnya adalah menghubungkan antar node dengan menghubungkan lapisan aplikasi. Langkah ini perlu dilakukan karena lapisan *transport* tidak dapat berdiri sendiri, maka perlu didefinisikan trafik yang dihubungkan dengan TCP. Oleh karena itu diperlukan

lapisan aplikasi. Pembangkitan trafik ini dilakukan dengan menuliskan kode program Tcl pada gambar 3.7

```
#=====
#           Applications Definition
#=====
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$cbro attach-agent $tcp0
$cbro set packetSize_ 32
$cbro set rate_ 0.1Mb
$cbro set random_
$ns at 0.001 "$cbro start"
$ns at 0.002 "$cbro stop"

#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$cbro1 attach-agent $tcp1
$cbro1 set packetSize_ 48
$cbro1 set rate_ 0.1Mb
$cbro1 set random_
$ns at 0.002 "$cbro1 start"
$ns at 0.003 "$cbro1 stop"

#Setup a CBR Application over TCP connection
set cbr2 [new Application/Traffic/CBR]
$cbro2 attach-agent $tcp2
$cbro2 set packetSize_ 60
$cbro2 set rate_ 0.1Mb
$cbro2 set random_
$ns at 0.003 "$cbro2 start"
$ns at 0.004 "$cbro2 stop"
```

Gambar 3.7 Pengaturan *Application Layer*

Pengaturan dari agen aplikasi disesuaikan dengan sistem dari simulasi yang diinginkan. Lapisan aplikasi yang digunakan pada simulasi ini adalah *Constant Bit Rate* (CBR). Pemilihan parameter tersebut karena sistem pada jaringan ini dibangkitkan secara kontinyu dan bit rate yang dipakai pada simulasi ini konstan dari satu *node* *node* lainnya.

3.5 Skenario Simulasi

Skenario simulasi pada tugas akhir ini pertama kali dilakukan dengan cara membuat topologi jaringan terlebih dahulu. Topologi ini dibuat dengan menyesuaikan topologi penerangan jalan umum. Setelah itu topologi tersebut divariasikan dengan jumlah node yang berbeda-

beda, mulai dari 3 node, 5 node, 7 node, 9 node. Topologi pada simulasi ini adalah topologi *point-to-point*. Hasil dari simulasi ini bertujuan untuk mengetahui batas terminal untuk melakukan komunikasi kooperatif pada sistem *smart lighting*. Setelah didapat nilai maksimal untuk melakukan komunikasi kooperatif, dilakukan pengujian mengenai kualitas layanan sistem tersebut dengan menganalisa nilai dari *packet loss ratio* dan nilai dari *end-to-end delay*. Setelah itu dilakukan simulasi kedua dengan cara membuat topologi *point-to-multipoint* untuk mengetahui *routing* yang dilakukan pada sistem *smart lighting* pada sistem secara keseluruhan. Simulasi dilakukan dengan tujuan mengetahui rute lain yang dilewati, sehingga sistem pada *smart lighting* memiliki *backup route* ketika salah satu rute tidak dapat dilewati. Node pada topologi ini berjumlah 17 *node*, dengan jumlah node 2×8 *node* sebagai *node* terminal dan satu *node* sebagai *node* pusat. Topologi ini tetap menyesuaikan dengan topologi pada penerangan jalan umum.

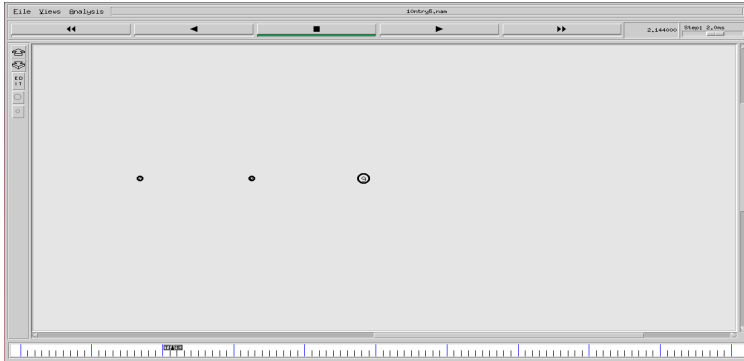
3.5.1 Pembuatan Topologi Jaringan

Pembuatan topologi jaringan pada simulasi ini menggunakan topologi *point-to-point* dan *point-to-multipoint*. Tingkat teratas dari topologi ini adalah Node pusat sebagai pusat pengumpulan data. Pada node tersebut diberikan program kode sebagai *sink* pada *transport agent*. Lalu node-node terminal sebagai node sensor akan mengirimkan data secara kooperatif untuk mencapai ke node pusat. Node-node tersebut diberikan perintah kode program sebagai TCP pada *transport agent*. Simulasi diawali dengan topologi *point-to-point* untuk mengetahui nilai kinerja jaringan dengan bertambahnya jumlah *node* serta perbedaan jarak antar *node*. Setelah itu dilakukan simulasi dengan topologi *point-to-multipoint* untuk mengetahui performa *routing* dari sistem komunikasi kooperatif pada *smart lighting*. Posisi dari *node* pusat berada pada sisi paling kanan, Sedangkan *node* sumber berada pada posisi paling kiri.

3.5.1.1 Topologi Jaringan dengan 3 node

Pada skenario simulasi ini menggunakan 3 *node*. 1 *node* berfungsi sebagai *node* pusat dan 2 *node* sebagai *node* terminal. Kondisi ini mewakili bentuk Penerangan Jalan Umum yang paling sederhana. Sehingga kondisi ini berfungsi untuk mengetahui kinerja jaringan pada

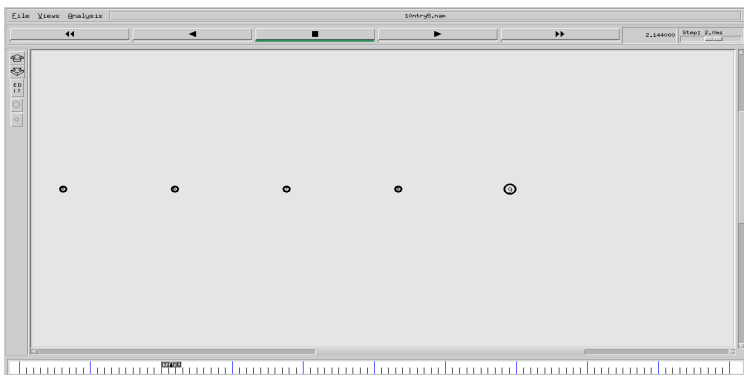
sistem yang paling sederhana. Gambar dari bentuk topologi pada simulasi jaringan ini ditunjukkan pada gambar 3.8.



Gambar 3.8 Topologi Jaringan 3 Node

3.5.1.2 Topologi Jaringan dengan 5 node

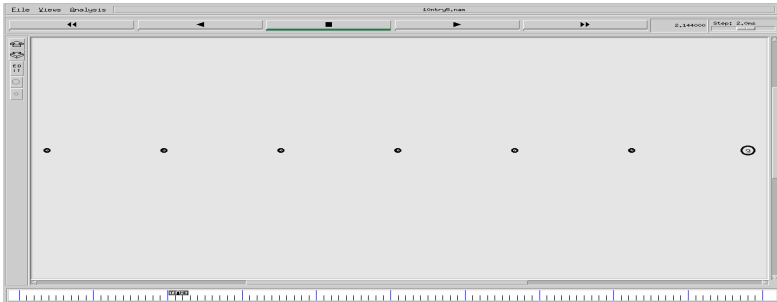
Topologi Selanjutnya adalah menggunakan skenario dengan jumlah 5 node, yang terdiri dari 4 node berfungsi sebagai node terminal dan 1 node sebagai node pusat. Penambahan jumlah node terminal ini bertujuan untuk mengetahui penambahan jumlah node terminal terhadap kinerja jaringan. Gambar dari bentuk topologi pada simulasi jaringan ini ditunjukkan pada gambar 3.9.



Gambar 3.9 Topologi Jaringan 5 Node

3.5.1.3 Topologi Jaringan dengan 7 node

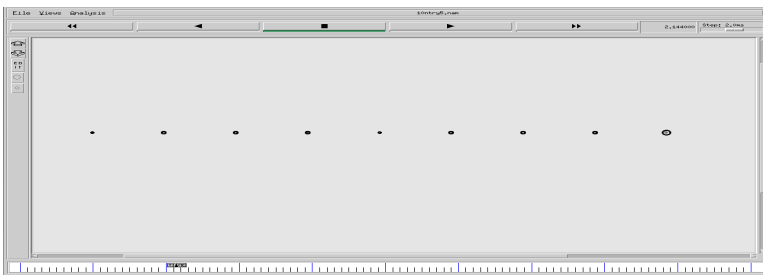
Topologi Selanjutnya adalah menggunakan skenario dengan jumlah 7 *node*, yang terdiri dari 6 *node* berfungsi sebagai *node* terminal dan 1 *node* sebagai *node* pusat. Penambahan jumlah *node* terminal ini bertujuan untuk mengetahui penambahan jumlah *node* terminal terhadap kinerja jaringan. Gambar dari bentuk topologi pada simulasi jaringan ini ditunjukkan pada gambar 3.10.



Gambar 3.10 Topologi Jaringan 7 Node

3.5.1.4 Topologi Jaringan dengan 9 node

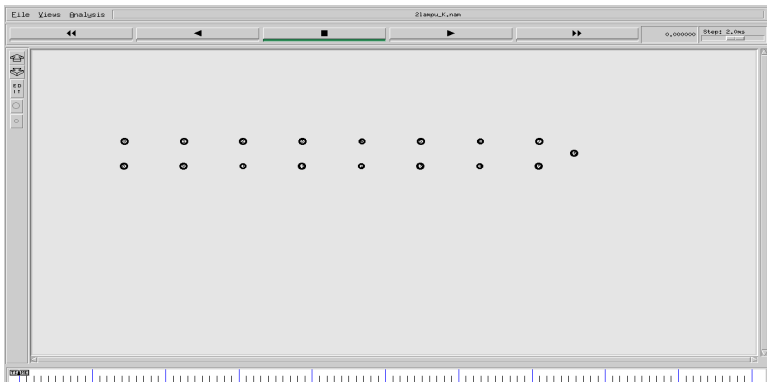
Topologi Selanjutnya adalah menggunakan skenario dengan jumlah 9 *node*, yang terdiri dari 9 *node* berfungsi sebagai *node* terminal dan 1 *node* sebagai *node* pusat. Penambahan jumlah *node* terminal ini bertujuan untuk mengetahui penambahan jumlah *node* terminal terhadap kinerja jaringan. Gambar dari bentuk topologi pada simulasi jaringan ini ditunjukkan pada gambar 3.11.



Gambar 3.11 Topologi Jaringan 9 Node

3.5.1.5 Topologi Jaringan *Point-to-Multipoint*

Pada skenario simulasi ini menggunakan jumlah 17 *node*. Nilai ini terdiri dari 2x8 *node* sebagai *node* terminal, dan 1 *node* sebagai *node* pusat. Jumlah *node* terminal hanya ditambahkan dua kalinya dari simulasi topologi *point-to-point*. Tujuan dari simulasi ini adalah untuk mengetahui performa *routing* dari jaringan kooperatif pada sistem *smart lighting*. Gambar dari bentuk topologi pada simulasi jaringan ini ditunjukkan pada gambar 3.12. Posisi urutan *node* pada simulasi ini tidak jauh berbeda dengan topologi sebelumnya. Posisi *node* pusat (*destination*) berada pada sisi paling kanan yaitu *node* 16. Sedangkan *node* terminal bagian atas adalah *node* 0,1,2,3,4,5,6,7 dan *node* terminal bagian bawah adalah *node* 8,9,10,11,12,13,14,15.



Gambar 3.12 Topologi Jaringan *Point-to-Multipoint*

Proses pengiriman data pada topologi ini dapat melalui jalur atau rute yang berbeda-beda pilihan rute yang dapat dilewati oleh data dari setiap *node* terminal menuju *node* pusat memiliki opsi lebih dari satu rute. Hal ini bertujuan untuk mengetahui kinerja dari jaringan yang sesungguhnya pada topologi penerangan lampu jalan dan untuk mengantisipasi jika rute dari salah satu jalur terputus, maka informasi tetap dapat menuju ke *destination* melalui jalur lain.

[Halaman ini sengaja dikosongkan]

BAB 4

PENGUJIAN DAN ANALISIS

4.1 Pendahuluan

Pada bab ini dijelaskan analisa sistem berdasarkan data yang diperoleh dari simulasi dan pengujian sistem yang telah dilakukan pada bab 3 mengenai simulasi komunikasi kooperatif pada *smart lighting*. Sistem dirancang menggunakan parameter simulasi yang bekerja sesuai dengan karakteristik pada sistem *smart lighting* yaitu menggunakan standar IEEE 802.15.4 dan menggunakan model propagasi *shadowing* untuk memberikan kondisi simulasi seperti kondisi yang sebenarnya. Hasil dari simulasi yang dilakukan kemudian akan dibandingkan nilai dari rata-rata *end to end delay*, *packet sent*, *packet drop*, *packet loss* dan nilai *throughput* yang telah dihasilkan dari hasil simulasi pada *software Network Simulator – 2*.

4.2 Performa Komunikasi Kooperatif *Point-to-Point*

Hasil dari simulasi yang telah dijalankan akan memunjukkan performa dari sistem komunikasi kooperatif berupa informasi mengenai nilai paket dan waktu dari pengiriman paket. Sistem ini dijalankan dengan jarak antar *node* yaitu 30, 40 dan 50 meter mengikuti standar dari penerangan jalan umum. Jumlah dari *node* yang bekerja berbeda-beda dimulai dari 3 *node*, 5 *node*, 7 *node* dan 9 *node* pada topologi *point-to-point*. Seteleah itu dilakukan percobaan dengan menggunakan topologi *point-to-multipoint* untuk mengetahui rute lain yang dapat dilalui. Berikut merupakan hasil data dari simulasi yang telah dijalankan dimulai dari jarak terendah dan jumlah *node* terkecil hingga jarak terjauh dengan dengan jumlah *node* terbanyak.

4.2.1 Performa Pada Jarak 30 Meter

Data pertama dari hasil simulasi adalah performa komunikasi kooperatif pada jarak 30 meter. Data yang dihasilkan dari simulasi menggunakan Network Simulator-2 ini adalah nilai dari jumlah paket secara keseluruhan, paket terkirim, paket yang di *drop*, paket yang hilang (*packet loss*), ukuran paket terkecil (Bytes), ukuran paket terbesar (Bytes), rata-rata ukuran paket (Bytes), jumlah paket terkirim (Bytes), jumlah paket yang di *drop* (Bytes), dan rata-rata *end-to-end delay*.

4.2.1.1 Performa Sistem Pada Topologi 3 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 3 *node* dengan jarak antar *node* yaitu 30 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 2 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, dan *node* 2. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 2 dan yang bertindak sebagai *node* terminal adalah *node* 0 dan *node* 1. Tabel 4.1 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	83 <i>packets</i>
<i>Number of Sent Packets</i>	82 <i>packets</i>
<i>Number of Dropped Packets</i>	3 <i>packets</i>
<i>Number of Loss Packets</i>	17 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	148 <i>packets</i>
<i>Average Packet Size</i>	50.49 <i>packets</i>
<i>Number of Sent Bytes</i>	4630 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	190 <i>Bytes</i>
<i>End to end Average Delay</i>	0.06 <i>sec</i>

Tabel 4.1 Informasi Simulasi pada Topologi 3 *Node* (30 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada jaringan dengan jarak 30 meter ini masih termasuk kecil yaitu berjumlah 82 paket. Nilai *dropped packets* dan *loss packet* juga tergolong kecil pada jaringan ini yaitu bernilai 3 paket pada *dropped packets* dan 17 paket pada *loss packets*. Sedangkan nilai rata-rata delay pada simulasi ini menghasilkan nilai 0.06 detik.

4.2.1.2 Performa Sistem Pada Topologi 5 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 5 *node* dengan jarak antar *node* yaitu 30 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 4 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3 dan *node* 4. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 4 dan

yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3. Tabel 4.2 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	236 <i>packets</i>
<i>Number of Sent Packets</i>	233 <i>packets</i>
<i>Number of Dropped Packets</i>	29 <i>packets</i>
<i>Number of Loss Packets</i>	61 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	176 <i>packets</i>
<i>Average Packet Size</i>	51.943 <i>packets</i>
<i>Number of Sent Bytes</i>	13414 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	1830 <i>Bytes</i>
<i>End to end Average Delay</i>	1.11 <i>sec</i>

Tabel 4.2 Informasi Simulasi pada Topologi 5 *Node* (30 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 5 *node*, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 5 *node* berjumlah 233 paket. Selain itu nilai dari *dropped packets* dan *loss packet*-nya pun bernilai semakin besar yaitu bernilai 29 *packet* pada *dropped packets* dan 61 *packet* pada *loss packets*. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 1.1 detik, Nilai ini cukup besar dibandingkan dengan nilai *delay* pada topologi sebelumnya.

4.2.1.3 Performa Sistem Pada Topologi 7 *Node*

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 7 *node* dengan jarak antar *node* yaitu 30 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 4 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3, *node* 4, *node* 5, *node* 6. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 6 dan yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3,4,5. Tabel 4.3 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	267 packets
<i>Number of Sent Packets</i>	266 packets
<i>Number of Dropped Packets</i>	29 packets
<i>Number of Loss Packets</i>	79 packets
<i>Minimal Packet Size</i>	28 packets
<i>Maximal Packet Size</i>	196 packets
<i>Average Packet Size</i>	51.803 packets
<i>Number of Sent Bytes</i>	15578 Bytes
<i>Number of Dropped Bytes</i>	2370 Bytes
<i>End to end Average Delay</i>	1.88 sec

Tabel 4.3 Informasi Simulasi pada Topologi 7 Node (30 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 7 node, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 7 node berjumlah 266 paket, nilai ini ada penambahan sedikit dibandingkan topologi sebelumnya. Selain itu nilai dari dropped packets pada jaringan ini berjumlah sama dari topologi sebelumnya yaitu 29 paket, namun nilai *loss packet* tetap bertambah menjadi 79 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 1.88 detik, Nilai ini bertambah dibandingkan dengan nilai *delay* pada topologi sebelumnya.

4.2.1.4 Performa Sistem Pada Topologi 9 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 9 *node*. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai node pusat, dan 4 *node* sebagai node terminal. *Node* tersebut dimulai dari *node 0, node 1, node 2, node 3, node 4, node 5, node 6, node 7, node 8*. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node 8* dan yang bertindak sebagai *node* terminal adalah *node 0,1,2,3,4,5,6,7*. Tabel 4.4 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	634 <i>packets</i>
<i>Number of Sent Packets</i>	620 <i>packets</i>
<i>Number of Dropped Packets</i>	86 <i>packets</i>
<i>Number of Loss Packets</i>	147 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	262 <i>packets</i>
<i>Average Packet Size</i>	53.78 <i>packets</i>
<i>Number of Sent Bytes</i>	37122 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	6180 <i>Bytes</i>
<i>End to end Average Delay</i>	3.84 <i>sec</i>

Tabel 4.4 Informasi Simulasi pada Topologi 9 *Node* (30 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 9 *node*, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada jumlah paket terkirim pada topologi 9 *node* berjumlah 620 paket, nilai ini bertambah cukup besar dibandingkan topologi sebelumnya. Selain itu nilai dari *dropped packets* pada jaringan ini juga bertambah dari topologi sebelumnya yaitu menjadi 86 paket dan nilai *loss packet* bertambah menjadi 147 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 3.84 detik, Nilai ini bertambah cukup besar dibandingkan dengan nilai *delay* pada topologi sebelumnya.

4.2.2 Performa Pada Jarak 40 Meter

Hasil kedua yang dari simulasi ini adalah performa komunikasi kooperatif pada jarak 40 meter. Data yang dihasilkan dari simulasi menggunakan Network Simulator-2 ini adalah nilai dari jumlah paket secara keseluruhan, paket terkirim, paket yang di *drop*, paket yang hilang (*packet loss*), ukuran paket terkecil (Bytes) , ukuran paket terbesar (Bytes), rata-rata ukuran paket (Bytes), jumlah paket terkirim (Bytes), jumlah paket yang di *drop* (Bytes), dan rata-rata *end-to-end delay*.

4.2.2.1 Performa Sistem Pada Topologi 3 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 3 *node* dengan jarak antar *node* yaitu 40 meter. Jumlah *node*

ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 2 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, dan *node* 2. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 2 dan yang bertindak sebagai *node* terminal adalah *node* 0 dan *node* 1. Tabel 4.5 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	68 <i>packets</i>
<i>Number of Sent Packets</i>	68 <i>packets</i>
<i>Number of Dropped Packets</i>	1 <i>packets</i>
<i>Number of Loss Packets</i>	3 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	148 <i>packets</i>
<i>Average Packet Size</i>	49.92 <i>packets</i>
<i>Number of Sent Bytes</i>	3990 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	40 <i>Bytes</i>
<i>End to end Average Delay</i>	0.03 <i>sec</i>

Tabel 4.5 Informasi Simulasi pada Topologi 3 *Node* (40 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada jaringan dengan jarak 40 meter ini bernilai 61 paket. Nilai *dropped packets* dan *loss packet* juga tergolong kecil pada jaringan ini yaitu bernilai 1 paket pada *dropped packets* dan 3 paket pada *loss packets*. Sedangkan nilai rata-rata delay pada simulasi ini menghasilkan nilai 0.03 detik. Jika dibandingkan dengan nilai paket yang terkirim pada topologi *node* yang sama yang berjumlah 3 *node* pada jarak 30 meter, paket terkirim pada jaringan ini berjumlah lebih sedikit.

4.2.2.2 Performa Sistem Pada Topologi 5 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 5 *node* dengan jarak antar *node* yaitu 40 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 4 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3 dan *node* 4. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 4 dan yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3. Tabel 4.6

akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	148 <i>packets</i>
<i>Number of Sent Packets</i>	148 <i>packets</i>
<i>Number of Dropped Packets</i>	22 <i>packets</i>
<i>Number of Loss Packets</i>	6 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	176 <i>packets</i>
<i>Average Packet Size</i>	51.53 <i>packets</i>
<i>Number of Sent Bytes</i>	8670 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	1578 <i>Bytes</i>
<i>End to end Average Delay</i>	0.06 <i>sec</i>

Tabel 4.6 Informasi Simulasi pada Topologi 5 Node (40 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 5 node, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 5 node berjumlah 148 paket. Selain itu nilai dari dropped packets dan loss packet-nya pun bernilai semakin besar yaitu bernilai 22 packet pada *dropped packets* dan 6 packet pada *loss packets*. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 0.06 detik, Nilai ini bertambah sedikit dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai paket yang terkirim pada topologi node yang sama sebelumnya yaitu berjumlah 5 node pada jarak 30 meter, paket terkirim pada jaringan ini juga berjumlah lebih sedikit.

4.2.2.3 Performa Sistem Pada Topologi 7 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 7 *node* dengan jarak antar node yaitu 40 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai node pusat, dan 4 *node* sebagai node terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3, *node* 4, *node* 5, *node* 6. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 6 dan yang bertindak sebagai *node* terminal adalah *node*

0,1,2,3,4,5. Tabel 4.7 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	284 <i>packets</i>
<i>Number of Sent Packets</i>	277 <i>packets</i>
<i>Number of Dropped Packets</i>	99 <i>packets</i>
<i>Number of Loss Packets</i>	6 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	212 <i>packets</i>
<i>Average Packet Size</i>	56.54 <i>packets</i>
<i>Number of Sent Bytes</i>	17204 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	8004 <i>Bytes</i>
<i>End to end Average Delay</i>	0.82 <i>sec</i>

Tabel 4.7 Informasi Simulasi pada Topologi 7 Node (40 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 7 node, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 7 node berjumlah 277 paket, nilai ini ada penambahan dibandingkan topologi sebelumnya. Selain itu nilai dari *loss packets* pada jaringan ini berjumlah sama dari topologi sebelumnya yaitu 6 paket, sedangkan nilai *drop packet* bertambah menjadi 99 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 0.82 detik, Nilai ini bertambah dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai paket yang terkirim pada topologi node yang sama sebelumnya yaitu berjumlah 7 node pada jarak 30 meter, paket terkirim pada jaringan ini tidak berbeda jauh namun sedikit lebih besar.

4.2.2.4 Performa Sistem Pada Topologi 9 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 9 *node* dengan jarak antar *node* yaitu 40 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai node pusat, dan 4 *node* sebagai node terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3, *node* 4, *node* 5, *node* 6, *node* 7, *node* 8. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 8 dan yang bertindak sebagai *node* terminal adalah

node 0,1,2,3,4,5,6,7. Tabel 4.8 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	534 <i>packets</i>
<i>Number of Sent Packets</i>	522 <i>packets</i>
<i>Number of Dropped Packets</i>	179 <i>packets</i>
<i>Number of Loss Packets</i>	38 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	262 <i>packets</i>
<i>Average Packet Size</i>	55.14 <i>packets</i>
<i>Number of Sent Bytes</i>	31318 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	14426 <i>Bytes</i>
<i>End to end Average Delay</i>	1.84 <i>sec</i>

Tabel 4.8 Informasi Simulasi pada Topologi 9 Node (40 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 9 node, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada jumlah paket terkirim pada topologi 9 node berjumlah 522 paket, nilai ini bertambah cukup besar dibandingkan topologi sebelumnya. Selain itu nilai dari dropped packets pada jaringan ini juga bertambah dari topologi sebelumnya yaitu menjadi 179 paket dan nilai *loss packet* bertambah menjadi 38 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 3.84 detik, Nilai ini bertambah cukup besar dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai paket yang terkirim pada topologi node yang sama sebelumnya yaitu berjumlah 9 node pada jarak 30 meter, paket terkirim pada jaringan ini juga berjumlah lebih sedikit

4.2.3 Performa Pada Jarak 50 Meter

Data pertama yang dianalisa adalah performa komunikasi kooperatif pada jarak 50 meter. Data yang dihasilkan dari simulasi menggunakan Network Simulator-2 ini adalah nilai dari jumlah paket secara keseluruhan, paket terkirim, paket yang di *drop*, paket yang hilang, ukuran paket terkecil (Bytes) , ukuran paket terbesar (Bytes),

rata-rata ukuran paket (Bytes), jumlah paket terkirim (Bytes), jumlah paket yang di drop (Bytes), dan rata-rata *end-to-end delay*.

4.2.3.1 Performa Sistem Pada Topologi 3 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 3 *node* dengan jarak antar *node* yaitu 50 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai node pusat, dan 2 *node* sebagai node terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, dan *node* 2. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 2 dan yang bertindak sebagai *node* terminal adalah *node* 0 dan *node* 1. Tabel 4.9 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	66 <i>packets</i>
<i>Number of Sent Packets</i>	66 <i>packets</i>
<i>Number of Dropped Packets</i>	1 <i>packets</i>
<i>Number of Loss Packets</i>	1 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	148 <i>packets</i>
<i>Average Packet Size</i>	50.06 <i>packets</i>
<i>Number of Sent Bytes</i>	3916 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	40 <i>Bytes</i>
<i>End to end Average Delay</i>	0.01 <i>sec</i>

Tabel 4.9 Informasi Simulasi pada Topologi 3 *Node* (50 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada jaringan dengan jarak 50 meter ini bernilai 66 paket. Nilai *dropped packets* dan *loss packet* juga tergolong kecil pada jaringan ini yaitu bernilai 1 paket pada *dropped packets* dan 1 paket pada *loss packets*. Sedangkan nilai rata-rata delay pada simulasi ini menghasilkan nilai 0.01 detik. Jika dibandingkan dengan nilai paket yang terkirim pada topologi *node* yang sama dan berjumlah 3 *node* pada jarak 40 meter, paket terkirim pada jaringan ini tidak jauh berbeda namun sedikit lebih besar.

4.2.3.2 Performa Sistem Pada Topologi 5 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 5 *node* dengan jarak antar *node* yaitu 50 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 4 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3 dan *node* 4. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 4 dan yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3. Tabel 4.10 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	145 <i>packets</i>
<i>Number of Sent Packets</i>	145 <i>packets</i>
<i>Number of Dropped Packets</i>	24 <i>packets</i>
<i>Number of Loss Packets</i>	1 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	176 <i>packets</i>
<i>Average Packet Size</i>	54.275 <i>packets</i>
<i>Number of Sent Bytes</i>	9042 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	2106 <i>Bytes</i>
<i>End to end Average Delay</i>	0.08 <i>sec</i>

Tabel 4.10 Informasi Simulasi pada Topologi 5 *Node* (50 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 5 *node*, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 5 *node* berjumlah 145 paket. Selain itu nilai dari *dropped packets* bernilai semakin besar yaitu bernilai 24 *packet*. Nilai dari *packet loss* bernilai sama dengan topologi sebelumnya yaitu 1 *packet*. Berarti dalam penambahan jumlah *node* pada simulasi ini tidak ada *packet* hilang yang baru. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 0.08 detik, Nilai ini bertambah sedikit dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai *packet* yang terkirim pada topologi *node* yang sama sebelumnya yaitu berjumlah 5 *node* pada jarak 30 meter, *packet* terkirim pada jaringan ini tidaklah jauh berbeda, namun bernilai lebih kecil.

4.2.3.3 Performa Sistem Pada Topologi 7 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 7 *node* dengan jarak antar *node* yaitu 50 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai *node* pusat, dan 4 *node* sebagai *node* terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3, *node* 4, *node* 5, *node* 6. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 6 dan yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3,4,5. Tabel 4.7 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Simulation Information</i>	
<i>Number of Generated Packets</i>	336 <i>packets</i>
<i>Number of Sent Packets</i>	332 <i>packets</i>
<i>Number of Dropped Packets</i>	76 <i>packets</i>
<i>Number of Loss Packets</i>	14 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	212 <i>packets</i>
<i>Average Packet Size</i>	54.01 <i>packets</i>
<i>Number of Sent Bytes-</i>	20444 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	5776 <i>Bytes</i>
<i>End to end Average Delay</i>	0.80 <i>sec</i>

Tabel 4.11 Informasi Simulasi pada Topologi 7 Node (50 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 7 *node*, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada paket terkirim pada topologi 7 *node* berjumlah 332 paket, nilai ini ada penambahan yang cukup besar dibandingkan topologi sebelumnya. Selain itu nilai dari *loss packets* pada jaringan ini bertambah dari topologi sebelumnya yaitu menjadi 15 paket, sedangkan nilai *drop packet* bertambah menjadi 76 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 0.80 detik, Nilai ini bertambah dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai paket yang terkirim pada topologi *node* yang sama sebelumnya yaitu berjumlah 7 *node* pada jarak 40 meter, paket terkirim pada jaringan ini bernilai lebih kecil.

4.2.3.4 Performa Sistem Pada Topologi 9 Node

Hasil performa sistem kooperatif ini menggunakan jumlah *node* sebanyak 9 *node* dengan jarak antar *node* 50 meter. Jumlah *node* ini mewakili jumlah dari terminal pada sistem *smart lighting*. Sistem ini terdiri dari 1 *node* sebagai node pusat, dan 4 *node* sebagai node terminal. *Node* tersebut dimulai dari *node* 0, *node* 1, *node* 2, *node* 3, *node* 4, *node* 5, *node* 6, *node* 7, *node* 8. Pada skenario ini yang bertindak sebagai *node* pusat adalah *node* 8 dan yang bertindak sebagai *node* terminal adalah *node* 0,1,2,3,4,5,6,7. Tabel 4.12 akan menunjukkan informasi mengenai hasil simulasi yang telah dijalankan.

<i>Number of Generated Packets</i>	354 packets
<i>Number of Sent Packets</i>	345 packets
<i>Number of Dropped Packets</i>	134 packets
<i>Number of Loss Packets</i>	28 packets
<i>Minimal Packet Size</i>	28 packets
<i>Maximal Packet Size</i>	262 packets
<i>Average Packet Size</i>	59.033 packets
<i>Number of Sent Bytes</i>	22854 Bytes
<i>Number of Dropped Bytes</i>	11660 Bytes
<i>End to end Average Delay</i>	0.8201 sec

Tabel 4.12 Informasi Simulasi pada Topologi 9 Node (50 meter)

Dari hasil simulasi dapat dilihat nilai dari paket terkirim pada topologi dengan jumlah 9 node, berjumlah lebih banyak dibandingkan dengan topologi sebelumnya. Nilai pada jumlah paket terkirim pada topologi 9 node berjumlah 345 paket, nilai ini bertambah dibandingkan topologi sebelumnya. Selain itu nilai dari *dropped packets* pada jaringan ini juga bertambah dari topologi sebelumnya yaitu menjadi 134 paket dan nilai *loss packet* bertambah menjadi 28 paket. Sedangkan rata-rata *end-to-end delay* pada simulasi ini bernilai 0.82 detik, Nilai ini bertambah sedikit dibandingkan dengan nilai *delay* pada topologi sebelumnya. Jika dibandingkan dengan nilai paket yang terkirim pada topologi node yang sama sebelumnya yaitu berjumlah 9 node pada jarak 40 meter, paket terkirim pada jaringan ini berjumlah lebih sedikit.

4.3 Analisa Performa Pada Seluruh Topologi *Point-to-Point*

Dari seluruh skenario simulasi komunikasi kooperatif pada sistem *smart lighting*, hasil dari simulasi tersebut dapat dibandingkan satu sama lainnya dari jumlah paket secara keseluruhan, *dropped packet*, *loss packet*, dan nilai *end-to-end delay*. Secara umum nilai dari paket secara keseluruhan, *dropped packet* dan *loss packet* semakin bertambah bersamaan dengan bertambahnya jumlah *node* pada sistem *smart lighting*. Nilai dari *end-to-end delay* juga semakin bertambah dengan bertambahnya jumlah *node*.

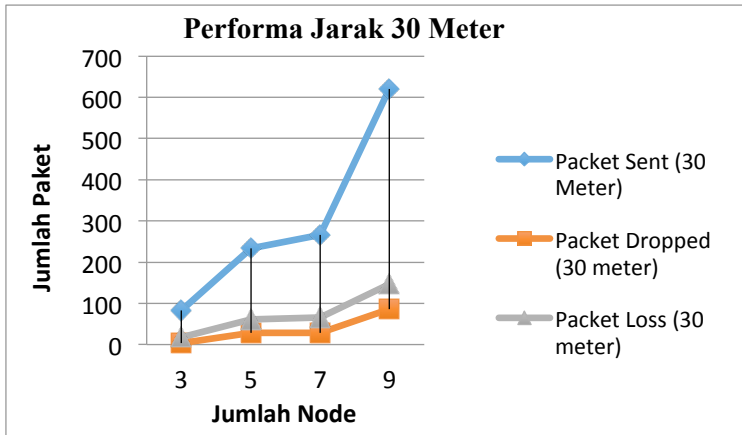
4.3.1 Analisa Performa Jaringan pada jarak 30 Meter

Parameter yang akan dianalisa pada jaringan kooperatif ini adalah pengaruh jumlah terminal atau *node* terhadap nilai *packet sent*, *end-to-end delay*, *packet drop*, *packet loss*. Tabel 4.14 akan memperlihatkan hasil dari performa jaringan kooperatif pada jarak antar *node* 30 meter dari seluruh topologi dimulai dari 3 *node*, 5 *node*, 7 *node* dan 9 *node*.

Kinerja Jaringan Kooperatif	Jumlah Node			
	3	5	7	9
<i>Packet Sent</i>	82	233	266	620
<i>Packet Dropped</i>	3	29	29	86
<i>Packet Loss</i>	17	61	66	147
<i>Avg delay (s)</i>	0.06	1.11	1.88	3.84

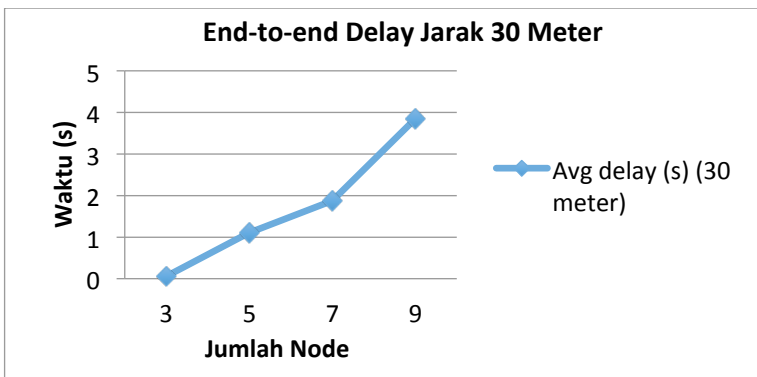
Tabel 4.13 Performa Jaringan Kooperatif Pada Jarak 30 Meter

Dari hasil performa jaringan kooperatif pada jumlah *node* yang berbeda-beda, dimulai dari *node* dengan jumlah terendah yaitu 3 *node* hingga 9 *node*. Nilai dari *packet loss* dan *packets dropped* juga semakin besar dengan bertambahnya jumlah paket yang dikirimkan. Selain itu nilai dari rata-rata *end-to-end delay* juga semakin bertambah bersamaan dengan bertambahnya jumlah *node*. Namun jika dilihat dari nilai paket *loss ratio* yang dihasilkan oleh performa jaringan ini, nilai yang dihasilkan dari setiap *node* bernilai besar yaitu dengan nilai rata-rata 20 %. Gambar 4.1 akan memperlihatkan grafik dari jumlah paket dengan bertambahnya jumlah *node*.



Gambar 4.1 Performa Jaringan Kooperatif pada Jarak 30 Meter

Secara Umum jumlah paket dari parameter yang dihasilkan seluruhnya bertambah, hal ini menunjukkan bahwa simulasi sesuai dengan teori pada sistem komunikasi kooperatif dimana nilai dari paket semakin besar dengan bertambahnya jumlah *node*. Selain jumlah paket, nilai dari rata-rata *end-to-end delay* pada simulasi ini juga semakin bertambah dengan bertambahnya jumlah node. Gambar 4.2 akan memperlihatkan grafik dari nilai *end-to-end delay* pada jaringan dengan jarak antar node 30 meter.



Gambar 4.2 Nilai *End-to-end Delay* pada Jarak 30 Meter

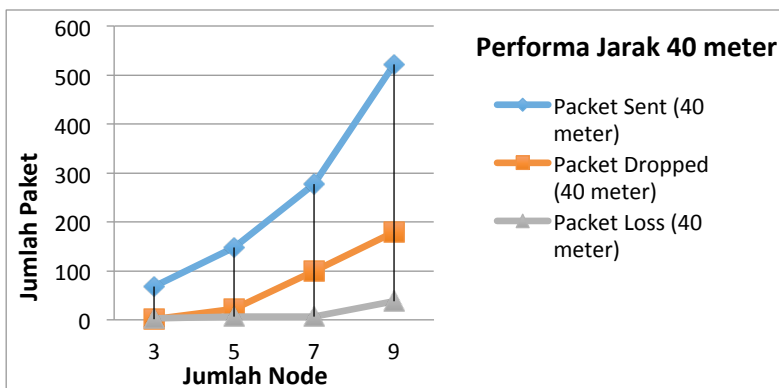
4.3.2 Analisa Performa Jaringan pada jarak 40 Meter

Parameter yang akan dianalisa pada jaringan kooperatif ini adalah pengaruh jumlah terminal atau *node* terhadap nilai *packet sent*, *end-to-end delay*, *packet drop*, *packet loss*. Tabel 4.14 akan memperlihatkan hasil dari performa jaringan kooperatif pada jarak antar node 40 meter dari seluruh topologi dimulai dari 3 *node* hingga 9 *node*.

Kinerja Jaringan Kooperatif	Jumlah Node			
	3	5	7	9
<i>Packet Sent</i>	68	148	277	522
<i>Packet Dropped</i>	1	22	99	179
<i>Packet Loss</i>	3	6	6	38
<i>Avg delay (s)</i>	0.03	0.06	0.82	1.84

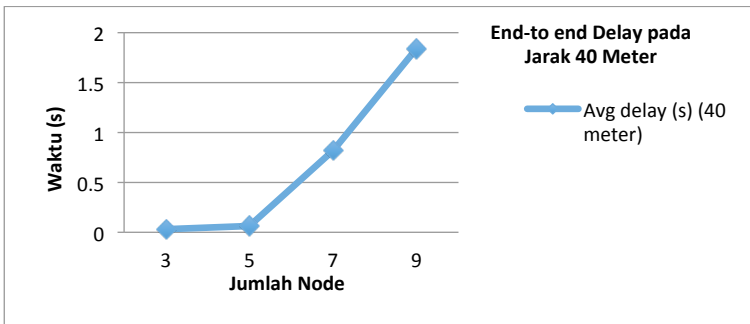
Tabel 4.14 Performa Jaringan Pada Jarak 40 Meter

Dari hasil performa jaringan kooperatif pada jumlah *node* yang berbeda-beda, dimulai dari *node* dengan jumlah terendah yaitu 3 *node* hingga 9 *node*. Nilai dari *packet loss* dan *packets dropped* juga semakin besar dengan bertambahnya jumlah paket yang dikirimkan kecuali pada *node* 7, nilai dari *packet loss* sama seperti pada *node* 5. Hal ini berarti pada proses pengiriman paket dari *node* 5 hingga *node* 7 tidak adanya paket yang hilang. Gambar 4.2 akan memperlihatkan grafik dari jumlah paket dengan bertambahnya jumlah node.



Gambar 4.3 Performa Jaringan Kooperatif pada Jarak 40 Meter

Secara Umum jumlah paket dari parameter yang dihasilkan seluruhnya bertambah, hal ini menunjukkan bahwa simulasi sesuai dengan teori pada sistem komunikasi kooperatif dimana nilai dari paket semakin besar dengan bertambahnya jumlah *node*. Namun jika dilihat dari nilai paket *loss ratio* yang dihasilkan oleh performa jaringan ini, nilai yang dihasilkan dari setiap *node* rata-rata bernilai 4 %. Selain jumlah paket, nilai dari rata-rata *end-to-end delay* pada simulasi ini juga semakin bertambah dengan bertambahnya jumlah *node*. Gambar 4.4 akan memperlihatkan grafik dari nilai *end-to-end delay* pada jaringan dengan jarak antar *node* 40 meter.



Gambar 4.4 Nilai *End-to-end Delay* pada Jarak 40 Meter

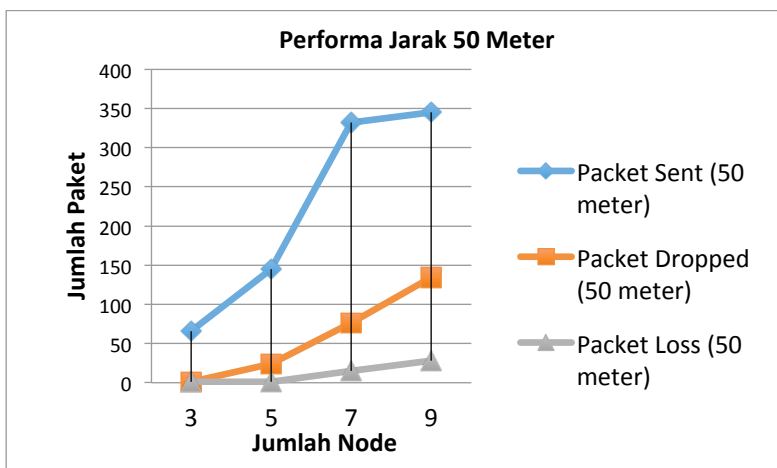
4.3.3 Analisa Performa Jaringan pada jarak 50 Meter

Parameter yang akan dianalisa pada jaringan kooperatif ini adalah pengaruh jumlah terminal atau *node* terhadap nilai *packet sent*, *end-to-end delay*, *packet drop*, *packet loss*. Tabel 4.14 akan memperlihatkan hasil dari performa jaringan kooperatif pada jarak antar *node* 40 meter dari seluruh topologi dimulai dari 3 *node* hingga 9 *node*.

Kinerja Jaringan Kooperatif	Jumlah Node			
	3	5	7	9
<i>Packet Sent</i>	66	145	332	345
<i>Packet Dropped</i>	1	24	76	134
<i>Packet Loss</i>	1	1	14	28
<i>Avg delay (s)</i>	0.01	0.08	0.80	0.820

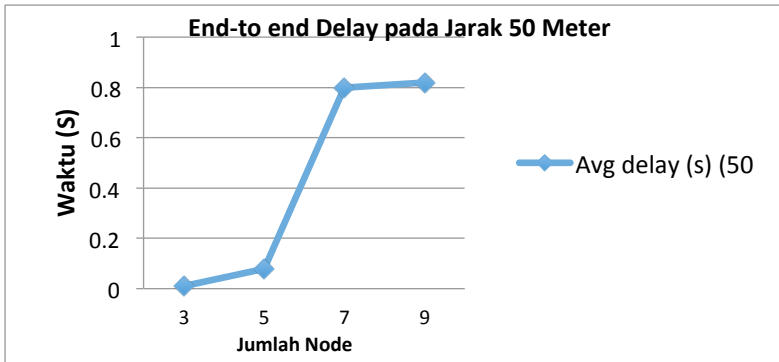
Tabel 4.15 Performa Jaringan Pada Jarak 50 Meter

Dari hasil performa jaringan kooperatif pada jumlah *node* yang berbeda-beda, dimulai dari *node* dengan jumlah terendah yaitu 3 *node* hingga 9 *node*. Nilai dari *packet loss* dan *packets dropped* juga semakin besar dengan bertambahnya jumlah paket yang dikirimkan. Selain itu nilai dari rata-rata *end-to-end* delay juga semakin bertambah bersamaan dengan bertambahnya jumlah *node*. Namun jika dilihat dari nilai paket *loss ratio* yang dihasilkan oleh performa jaringan ini, nilai yang dihasilkan dari setiap *node* bernilai lebih kecil dari performa pada jaringan sebelumnya yaitu bernilai nilai rata-rata 4 %. Gambar 4.5 akan memperlihatkan grafik dari jumlah paket dengan bertambahnya jumlah *node*.



Gambar 4.5 Performa Jaringan Kooperatif pada Jarak 50 Meter

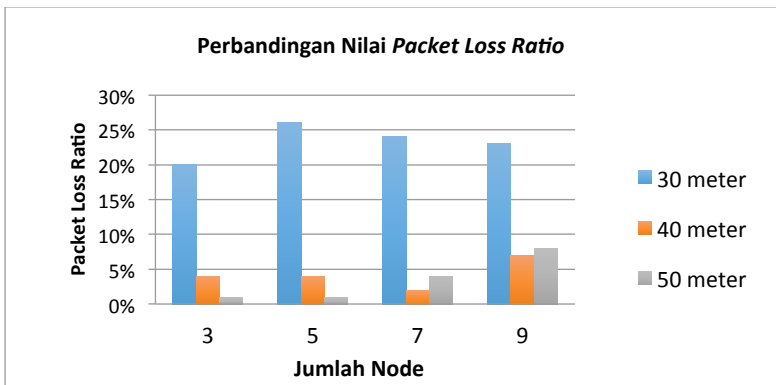
Secara Umum jumlah paket dari parameter yang dihasilkan seluruhnya bertambah, hal ini menunjukkan bahwa simulasi sesuai dengan teori pada sistem komunikasi kooperatif dimana nilai dari paket semakin besar dengan bertambahnya jumlah *node*. Selain jumlah paket, nilai dari rata-rata *end-to-end* delay pada simulasi ini juga semakin bertambah dengan bertambahnya jumlah *node*. Gambar 4.6 akan memperlihatkan grafik dari nilai *end-to-end* delay pada jaringan dengan jarak antar *node* 50 meter.



Gambar 4.6 Performa Jaringan Kooperatif pada Jarak 50 Meter

4.4 Perbandingan Performa Jaringan pada Seluruh Topologi *Point-to-Point*

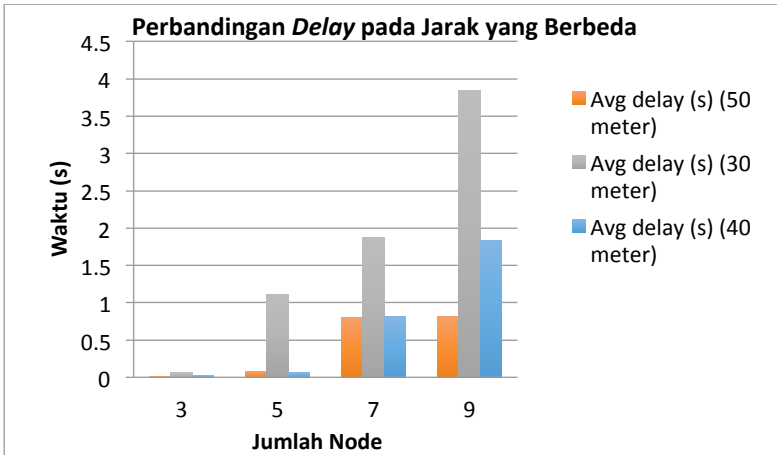
Setelah performa dari setiap topologi dianalisa dengan perbandingan dari jumlah *node*, diperlukan analisa mengenai perbandingannya pada jarak yang berbeda untuk menentukan jarak yang paling optimal dari sistem komunikasi kooperatif pada *smart lighting*. Kinerja paling baik dari sistem ini adalah pada jaringan dengan nilai *packet loss ratio* terendah dan memiliki nilai *delay* yang kecil. Nilai dari perbandingan *packet loss ratio* akan diperlihatkan pada grafik pada gambar 4.7.



Gambar 4.7 Performa Jaringan pada Topologi *Point-to-Point*

Nilai dari *packet loss ratio* terbesar berada pada jaringan topologi dengan jarak antar *node* 30 meter. Hal ini disebabkan karena tingkat kepadatan jaringan karena proses pengiriman informasi yang begitu tinggi sedangkan nilai dari *packet loss ratio* bernilai lebih kecil pada jaringan topologi dengan jarak antar *node* 40 dan 50 meter. Hal ini disebabkan paket yang dapat terkirim dari topologi ini bernilai lebih kecil dibandingkan dengan jarak yang lebih dekat, sehingga tingkat kepadatan pada jaringan ini tidak sepadat pada jaringan yang berjarak lebih jauh. Hal ini menyebabkan nilai dari paket yang hilang lebih kecil.

Selain nilai dari *packet loss ratio* nilai lain yang perlu dianalisa adalah nilai *end-to-end delay* dari jaringan kooperatif dengan perbedaan jarak antar *node*. Perbandingan dari nilai *end-to-end delay* pada jaringan kooperatif dengan topologi *point-to-point* dapat dilihat dari grafik yang ditunjukkan pada gambar 4.8



Gambar 4.8 Perbandingan End-to-end Delay pada Jarak yang Berbeda

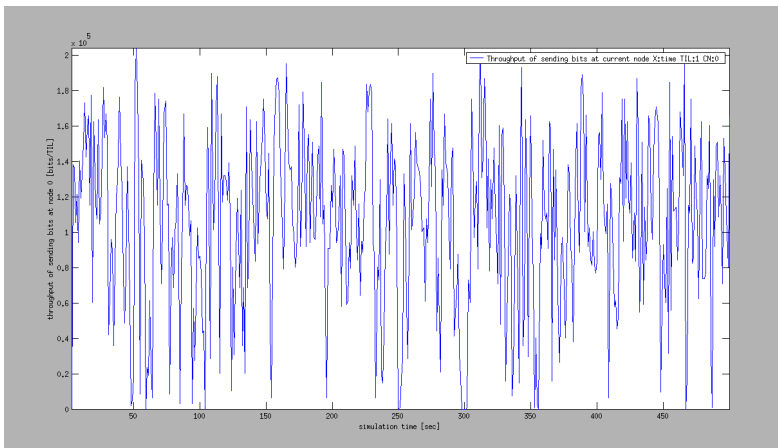
Perbandingan nilai *end-to-end delay* pada jaringan komunikasi kooperatif dengan jarak yang berbeda dihasilkan nilai dari *delay* terbesar berada pada jaringan dengan jarak antar *node* 30 meter. Sedangkan nilai terendah dari *end-to-end delay* berada pada jaringan dengan jarak antar *node* terjauh yaitu dengan jarak 50 meter. Hal ini disebabkan karena

kepadatan jaringan yang berbeda pada jaringan dengan jarak antar *node* yang berbeda. Nilai dari *end-to-end* delay pada jumlah node terbesar pada jarak ini bernilai berada dibawah 1 detik. Nilai ini masih dapat ditoleransi jika diterapkan untuk *real time monitoring*.

Dari hasil perbandingan jaringan pada jarak yang berbeda antar *node* dapat dilihat bahwa nilai dari *packet loss ratio* paling rendah dan nilai *end-to-end* paling rendah berada pada jaringan komunikasi kooperatif pada jarak antar *node* 50 meter.

4.5 Hasil Throughput

Dari seluruh simulasi yang telah dijalankan didapatkan nilai paling optimal dari topologi jaringan kooperatif yaitu dengan jarak antar *node* 50 meter. Sehingga jaringan ini diperlukan analisa mengenai nilai throughput yang dihasilkan. Nilai dari throughput yang dihasilkan dapat dilihat pada gambar 4.9.



Gambar 4.9 Hasil Throughput pada Jarak Antar *Node* 50 meter

Nilai dari throughput yang didapat dari hasil simulasi jaringan kooperatif pada sistem smart lighting dengan jarak antar *node* 50 meter bernilai rata-rata 1.3×10^5 bits/s atau bernilai 130 Kb/s.

4.6 Performa Jaringan Kooperatif *Point-to-Multipoint*

Kinerja dari jaringan kooperatif tidak dapat dianalisa hanya dengan topologi *point-to-point* dikarenakan topologi tersebut tidak dapat menunjukkan performa routing yang dilakukan pada jaringan kooperatif. Sehingga dilakukan pengujian performa routing dengan jarak yang memiliki QoS terbaik dan topologi *point-to-multipoint* agar dapat menganalisa kinerja routing yang dilakukan. Tabel 4.16 akan menunjukan hasil simulasi jaringan kooperatif pada topologi *point-to-multipoint*.

<i>Number of Generated Packets</i>	2160 <i>packets</i>
<i>Number of Sent Packets</i>	2106 <i>packets</i>
<i>Number of Dropped Packets</i>	1712 <i>packets</i>
<i>Number of Loss Packets</i>	188 <i>packets</i>
<i>Minimal Packet Size</i>	28 <i>packets</i>
<i>Maximal Packet Size</i>	390 <i>packets</i>
<i>Average Packet Size</i>	63.829 <i>packets</i>
<i>Number of Sent Bytes</i>	137556 <i>Bytes</i>
<i>Number of Dropped Bytes</i>	159678 <i>Bytes</i>
<i>Max Average Delay</i>	0.938 <i>sec</i>
<i>Min Average Delay</i>	0.812 <i>sec</i>

Tabel 4.16 Informasi Simulasi pada Topologi *Point-to-Multipoint*

Dari hasil simulasi didapatkan nilai dari paket terkirim cukup besar dikarenakan banyaknya jumlah node yang melakukan pengiriman paket sehingga menyebabkan nilai *dropped packet* juga tinggi. Namun nilai dari *packet loss ratio* pada topologi ini bernilai 8.7 %, nilai ini tidak jauh berbeda pada topologi *point-to-point*. Jika dilihat pada nilai *delay* nilai maksimal pada topologi ini bernilai 0.93 detik. Nilai ini juga tidak jauh berbeda jika dibandingkan topologi *point-to-point*, Hal ini disebabkan karena jumlah *node* yang dilalui pada jaringan kooperatif pada topologi *point-to-multipoint* bernilai sama dengan jaringan kooperatif pada topologi *point-to-point*. Hal ini bertujuan untuk menjaga nilai QoS pada jaringan ini agar *delay* dan nilai *packet loss ratio* tidak terlalu tinggi. Namun keunggulan jaringan kooperatif pada topologi *point-to-multipoint* adalah adanya *backup route* jika salah satu rute pada jaringan ini terjadi kendala dan menyebabkan terputusnya jaringan. Sehingga informasi dari setiap *node* lain tetap dapat terkirim menuju

terminal pusat (*destination*). Untuk informasi mengenai rute dari setiap *node* akan diperlihatkan pada tabel 4.17.

Source	Destination	Hops Count	Next Hops	sequence
0	16	8	1 / 9	1
1	16	7	2 / 10	2
2	16	6	3 / 9	3
3	16	5	4 / 12	4
4	16	4	5 / 13	5
5	16	3	6 / 14	6
6	16	2	7 / 15	7
7	16	1	16	8
8	16	8	1 / 9	1
9	16	7	2 / 10	2
10	16	6	3 / 9	3
11	16	5	4 / 12	4
12	16	4	5 / 13	5
13	16	3	6 / 14	6
14	16	2	7 / 15	7
15	16	1	16	8

Tabel 4.17 Informasi *Routing Table* dari Setiap *Node*

Pada jaringan kooperatif setiap *node* sudah memiliki *table routing* masing-masing untuk melakukan pengiriman informasi menuju terminal pusat, hal ini dikarenakan tidak adanya mobilitas pada setiap *node* sehingga tidak perlu adanya *update* untuk menentukan rute baru dikarenakan jalur pada jaringan yang akan dilewati sudah jelas. Dari informasi mengenai *routing table* pada table 4.17, nilai dari setiap *node* dapat mengirimkan informasi melalui 2 jalur yang berbeda menuju *destination* dengan urutan pengiriman dari *node* terjauh mengirim informasi terlebih dahulu hingga menuju *node* terdekat dari terminal pusat (*destination*).

[Halaman ini sengaja dikosongkan]

BAB 5

KESIMPULAN DAN SARAN

Setelah serangkaian pengujian dan perancangan yang telah dilakukan dianalisis maka akan dapat ditarik kesimpulan. Pembahasan dari bab-bab sebelumnya pada tugas akhir ini akan menjadi bahan pertimbangan atau referensi dalam melakukan penelitian pengembangan dari tugas akhir ini.

5.1 Kesimpulan

Berdasarkan hasil perencanaan dan hasil pengujian sistem komunikasi kooperatif pada *smart lighting* maka dapat ditarik beberapa kesimpulan, diantaranya :

1. Jumlah lampu yang mampu melakukan komunikasi kooperatif pada sistem smart lighting tidak dapat ditentukan secara pasti untuk jarak 30 meter dan 40 meter. Hal ini disebabkan karena nilai *packet loss ratio* dan *end-to-end delay* pada jaringan tersebut bernilai cukup tinggi. Sedangkan jumlah lampu yang mampu melakukan komunikasi kooperatif pada jarak 50 meter berjumlah 9 lampu. Jumlah ini dilihat dari nilai *packet loss ratio* yang rendah.
2. Bertambahnya jumlah *node* akan menyebabkan bertambahnya data yang akan dikirimkan. Hal ini disebabkan semakin besarnya informasi atau paket yang harus dikirimkan dari setiap *node*.
3. Bertambahnya jumlah *node* akan menyebabkan nilai dari *packet dropped* semakin besar, hal ini disebabkan karena semakin padatnya jaringan menyebabkan antrian untuk pengiriman paket semakin besar, sehingga dapat menyebabkan *timeout* sehingga paket gagal untuk dikirimkan dan akan di *dropped*.
4. Semakin bertambahnya jumlah *node* secara umum menyebabkan nilai dari *packet loss* bertambah.
5. Nilai dari *end-to-end delay* akan bertambah bersamaan dengan bertambahnya jumlah *node*. Hal ini disebabkan karena semakin padatnya jaringan dan waktu yang diperlukan proses pengiriman data semakin tinggi.

6. Performa jaringan pada topologi *point-to-multipoint* memiliki nilai *loss packet ratio* dan *delay* yang tidak jauh berbeda dengan topologi jaringan kooperatif pada topologi *point-to-multipoint*, hal ini dikarenakan jumlah *node* yang dilalui bernilai sama. Selain itu setiap *node* pada topologi *point-to-multipoint* memiliki *backup route* untuk mengantisipasi salah satu rute yang dilewati terjadi kendala.

5.2 Saran

Tugas akhir ini melakukan simulasi sistem komunikasi kooperatif pada *smart lighting* menggunakan Network Simulator -2, sehingga ada beberapa saran untuk penelitian selanjutnya :

1. Menambahkan kondisi lain sebagai variabel untuk melihat parameter lainnya dalam menganalisa kinerja jaringan
2. Menjalankan simulasi komunikasi kooperatif pada aplikasi network simulator -3 (NS-3) software dengan generasi terbaru.
3. Mencoba mengimplementasikan sistem ini dengan membuat alatnya pada jaringan sesungguhnya yaitu WSN (*Wireless Sensor Network*).

DAFTAR PUSTAKA

- [1] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," in *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22-32, Feb. 2014.
- [2] N. Ouerhani, N. Pazos, M. Aeberli and M. Muller, "IoT-based dynamic street light control for smart cities use cases," *2016 International Symposium on Networks, Computers and Communications (ISNCC)*, Yasmine Hammamet, 2016, pp. 1-5.
- [3] P. C. Veena, P. Tharakan, H. Haridas, K. Ramya, R. Joju and T. S. Jyothis, "Smart street light system based on image processing," *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, Nagercoil, 2016, pp. 1-5.
- [4] B. Abinaya, S. Gurupriya and M. Pooja, "Iot based smart and adaptive lighting in street lights," *2017 2nd International Conference on Computing and Communications Technologies (ICCCT)*, Chennai, 2017, pp. 195-198.
- [5] Z. Ke and C. Xiao, "Research of Intelligent Street Light System Based on ZigBee," *2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII)*, Wuhan, 2016, pp. 255-258.
- [6] K.J. Ray Liu, Ahmed K.Sadek, Weifeng Su and Andres Kwasinski, "Cooperative Communications and Networking", 2009.
- [7] P. Elejoste *et al.*, "An Easy to Deploy Street Light Control System Based on Wireless Communication and LED Technology," *Sensors*, vol. 13, no. 5, pp. 6492–6523, May 2013

- [8] P. Elejoste *et al.*, “An Easy to Deploy Street Light Control System Based on Wireless Communication and LED Technology,” *Sensors*, vol. 13, no. 5, pp. 6492–6523, May 2013.
- [9] P. Sriploy, P. Uthansakul, and M. Uthansakul, “Effect of path loss on the distributed beamforming for Wireless Sensor Networks,” 2013, pp. 1–4.
- [10] P. Stuedi, O. Chinellato, and G. Alonso, “Connectivity in the presence of shadowing in 802.11 ad hoc networks,” 2005, vol. 4, pp.2225–2230.
- [11] A. Moragrega and C. Ibars, “Performance analysis of cooperative and range based localization algorithms for Zigbee and 802.15.4a Wireless Sensor Networks,” 2010, pp. 1996–2001
- [12] S. Safaric and K. Malaric, “ZigBee wireless standard,” 2006, pp. 259–262.

LAMPIRAN

LEMBAR PENGESAHAN PROPOSAL TA

Departemen Teknik Elektro
Fakultas Teknologi Elektro - ITS

TE 141599 TUGAS AKHIR – 4 SKS

Nama Mahasiswa : Muhammad Ilham Pratomo
Nomer Pokok : 07111440000167
Bidang Studi : Telekomunikasi Multimedia
Tugas Diberikan : Semester Genap 2017/2018
Dosen Pembimbing : 1. Dr.Ir. Endroyono, DEA
2. Ir. Gatot Kusrahardjo, MT.

09 FEB 2018

Judul Tugas Akhir : **Evaluasi Komunikasi Kooperatif Pada Sistem Smart Lighting**
(*Evaluation of Cooperative Communication in Smart Lighting System*)

Uraian Tugas Akhir :

Smart lighting system merupakan salah satu sistem yang diterapkan pada salah satu pilar pada *smart city* yang berfungsi untuk mempermudah masyarakat maupun pemerintah kota dalam mengontrol sistem pencahayaan pada penerangan kota dan memonitoring kondisi yang terjadi pada lingkungan umum. Sistem *smart lighting* pada umumnya menerapkan sistem komunikasi kooperatif. Sistem ini diterapkan karena kestabilannya dalam hal efisiensi energi karena mengurangi daya transmisi serta dapat mengoptimalkan kapasitas data yang akan ditransmisikan. Kecerdasan pada kemampuan *smart lighting* bergantung pada sistem yang dibutuhkan suatu kota dalam menyelesaikan suatu permasalahan. Semakin banyak permasalahan yang ingin diatasi maka dibutuhkan kemampuan pada *smart lighting* yang semakin lengkap. Kemampuan ini bergantung pada jumlah *device* dan sensor yang digunakan. Semakin banyak sensor yang digunakan, maka semakin cerdas kemampuan pada *smart lighting* tersebut dan semakin banyak sensor yang digunakan, maka semakin besar juga jumlah data yang harus di transmisikan, Sehingga membutuhkan kapasitas yang besar. Oleh sebab itu pada tugas akhir ini akan dilakukan evaluasi kinerja pada sistem komunikasi kooperatif yang diterapkan pada sistem *smart lighting*. Parameter yang akan di evaluasi adalah kapasitas data yang terdiri dari *throughput*, *delay*, *jitter* serta daya pancar dan jarak antar terminal pada *smart lighting*.

Dosen Pembimbing I,



Dr.Ir. Endroyono, DEA
NIP: 196504041991021001

Dosen Pembimbing II,



Ir. Gatot Kusrahardjo, MT
NIP: 195904281986011001

Mengetahui,
Ketua Program Studi S1



Dedet C. Riawan, S.T., M.Eng., Ph.D.
NIP: 197311192000031001

Menyetujui,
Kepala Laboratorium Komunikasi
Multimedia



Dr.Ir. Endroyono, DEA
NIP: 196504041991021001

[Halaman ini sengaja dikosongkan]

LISTING PROGRAM

1. Simulasi Topologi Jaringan 3 Node

```
#####  
# Simulation parameters setup  
#####  
Antenna/OmniAntenna set Gt_1 ;#Transmit antenna gain  
Antenna/OmniAntenna set Gr_1 ;#Receive antenna gain  
Phy/WirelessPhy set pathlossExp_ 2  
Phy/WirelessPhy set std_db_ 12  
Phy/WirelessPhy set L_ 1.0 ;#System Loss Factor  
Phy/WirelessPhy set freq_ 2.452e9 ;#channel  
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate  
Phy/WirelessPhy set Pt_ 0.06 ;#Transmit Power  
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power Threshold  
Mac/802_11 set dataRate_ 0.1Mb ;#Rate for Data Frames  
Mac/802_11 set basicRate_ 0.05Mb ;#Rate for Control Frames  
  
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) 50 ;# max packet in ifq  
set val(nn) 3 ;# number of mobilenodes  
set val(rp) AODV ;# routing protocol  
set val(x) 697 ;# X dimension of topography  
set val(y) 500 ;# Y dimension of topography  
set val(stop) 10.0 ;# time of simulation end  
  
#####  
# Initialization  
#####  
#Create a ns simulator
```

```

set ns [new Simulator]

#Setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)

#Open the NS trace file
set tracefile [open 12lampu_Z.tr w]
$ns trace-all $tracefile

#Open the NAM trace file
set namfile [open 12lampu_Z.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel

#=====
#   Mobile node parameter setup
#=====
$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) \
    -channel $chan \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace OFF

#=====
#   Nodes Definition

```



```

#=====
#Create 3 nodes
set n0 [$ns node]
$n0 set X_ 197
$n0 set Y_ 400
$n0 set Z_ 0.0
$ns initial_node_pos $n0 20
set n1 [$ns node]
$n1 set X_ 397
$n1 set Y_ 400
$n1 set Z_ 0.0
$ns initial_node_pos $n1 20
set n2 [$ns node]
$n2 set X_ 597
$n2 set Y_ 400
$n2 set Z_ 0.0
$ns initial_node_pos $n2 20

#=====
#   Agents Definition
#=====
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink2 [new Agent/TCPSink]
$ns attach-agent $n1 $sink2
$ns connect $tcp0 $sink2
$tcp0 set packetSize_ 34

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink3 [new Agent/TCPSink]
$ns attach-agent $n2 $sink3
$ns connect $tcp1 $sink3
$tcp1 set packetSize_ 50

```

```

#=====
#   Applications Definition
#=====
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $tcp0
$cbr0 set packetSize_ 34
$cbr0 set rate_ 0.1Mb
$cbr0 set random_
$ns at 0.001 "$cbr0 start"
$ns at 0.002 "$cbr0 stop"

#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $tcp1
$cbr1 set packetSize_ 60
$cbr1 set rate_ 0.1Mb
$cbr1 set random_
$ns at 0.002 "$cbr1 start"
$ns at 0.003 "$cbr1 stop"

#=====
#   Termination
#=====
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
    close $tracefile
    close $namfile
    exec nam 12lampu_Z.nam &
    exit 0
}
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "\$n$i reset"
}

```

```

}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run

```

2. Simulasi Topologi Jaringan 5 Node

```

#=====
# Simulation parameters setup
#=====
Antenna/OmniAntenna set Gt_1 ;#Transmit antenna gain
Antenna/OmniAntenna set Gr_1 ;#Receive antenna gain
Phy/WirelessPhy set pathlossExp_2
Phy/WirelessPhy set std_db_12
Phy/WirelessPhy set L_1.0 ;#System Loss Factor
Phy/WirelessPhy set freq_ 2.452e9 ;#channel
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate
Phy/WirelessPhy set Pt_ 0.06 ;#Transmit Power
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power Threshold
Mac/802_15_4 set dataRate_ 0.1Mb ;#Rate for Data Frames
Mac/802_15_4 set basicRate_ 0.05Mb ;#Rate for Control
Frames

set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation
model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(nn) 3 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 702 ;# X dimension of topography

```

```
set val(y) 498 ;# Y dimension of topography
set val(stop) 10.0 ;# time of simulation end
```

```
#=====
# Initialization
#=====
```

```
#Create a ns simulator
set ns [new Simulator]
```

```
#Setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
```

```
#Open the NS trace file
set tracefile [open 3lampu_Z.tr w]
$ns trace-all $tracefile
```

```
#Open the NAM trace file
set namfile [open 3lampu_Z.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel
```

```
#=====
# Mobile node parameter setup
#=====
```

```
$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) \
    -channel $chan \
    -topoInstance $topo \
```

```
-agentTrace ON \  
-routerTrace ON \  
-macTrace ON \  
-movementTrace OFF
```

```
#=====
```

```
# Nodes Definition
```

```
#=====
```

```
#Create 3 nodes
```

```
set n0 [$ns node]
```

```
$n0 set X_ 202
```

```
$n0 set Y_ 398
```

```
$n0 set Z_ 0.0
```

```
$ns initial_node_pos $n0 20
```

```
set n1 [$ns node]
```

```
$n1 set X_ 402
```

```
$n1 set Y_ 398
```

```
$n1 set Z_ 0.0
```

```
$ns initial_node_pos $n1 20
```

```
set n2 [$ns node]
```

```
$n2 set X_ 602
```

```
$n2 set Y_ 398
```

```
$n2 set Z_ 0.0
```

```
$ns initial_node_pos $n2 20
```

```
#=====
```

```
# Agents Definition
```

```
#=====
```

```
#Setup a TCP connection
```

```
set tcp0 [new Agent/TCP]
```

```
$ns attach-agent $n0 $tcp0
```

```
set sink2 [new Agent/TCPSink]
```

```
$ns attach-agent $n1 $sink2
```

```
$ns connect $tcp0 $sink2
```

```
$tcp0 set packetSize_ 30
```

```
#Setup a TCP connection
```

```
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink3 [new Agent/TCPSink]
$ns attach-agent $n2 $sink3
$ns connect $tcp1 $sink3
$tcp1 set packetSize_ 60
```

```
#=====
#   Applications Definition
#=====
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $tcp0
$cbr0 set packetSize_ 30
$cbr0 set rate_ 0.1Mb
$cbr0 set random_
$ns at 0.1 "$cbr0 start"
$ns at 0.2 "$cbr0 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $tcp1
$cbr1 set packetSize_ 60
$cbr1 set rate_ 0.1Mb
$cbr1 set random_
$ns at 0.2 "$cbr1 start"
$ns at 0.3 "$cbr1 stop"
```

```
#=====
#   Termination
#=====
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
```

```

close $tracefile
close $namfile
exec nam 3lampu_Z.nam &
exit 0
}
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "\$n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run

```

3. Simulasi Topologi Jaringan 7 Node

```

#=====
# Simulation parameters setup
#=====
Antenna/OmniAntenna set Gt_1 ;#Transmit antenna gain
Antenna/OmniAntenna set Gr_1 ;#Receive antenna gain
Phy/WirelessPhy set pathlossExp_2
Phy/WirelessPhy set std_db_12
Phy/WirelessPhy set L_1.0 ;#System Loss Factor
Phy/WirelessPhy set freq_ 2.452e9 ;#channel
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate
Phy/WirelessPhy set Pt_ 0.06 ;#Transmit Power
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power Threshold
Mac/802_15_4 set dataRate_ 0.1Mb ;#Rate for Data Frames
Mac/802_15_4 set basicRate_ 0.05Mb ;#Rate for Control
Frames

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) 50 ;# max packet in ifq
set val(nn) 7 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 1501 ;# X dimension of topography
set val(y) 500 ;# Y dimension of topography
set val(stop) 10.0 ;# time of simulation end

```

```

#=====

```

```

# Initialization

```

```

#=====

```

```

#Create a ns simulator

```

```

set ns [new Simulator]

```

```

#Setup topography object

```

```

set topo [new Topography]

```

```

$topo load_flatgrid $val(x) $val(y)

```

```

create-god $val(nn)

```

```

#Open the NS trace file

```

```

set tracefile [open 7lampu_Z.tr w]

```

```

$ns trace-all $tracefile

```

```

#Open the NAM trace file

```

```

set namfile [open 7lampu_Z.nam w]

```

```

$ns namtrace-all $namfile

```

```

$ns namtrace-all-wireless $namfile $val(x) $val(y)

```

```

set chan [new $val(chan)];#Create wireless channel

```

```

#=====

```

```

# Mobile node parameter setup

```

```

#=====

```

```

$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) \  
-channel    $chan \  
-topoInstance $topo \  
-agentTrace ON \  
-routerTrace ON \  
-macTrace   ON \  
-movementTrace OFF
```

```
#=====\  
#   Nodes Definition  
#=====\  
#Create 7 nodes  
set n0 [$ns node]  
$n0 set X_ 201  
$n0 set Y_ 400  
$n0 set Z_ 0.0  
$ns initial_node_pos $n0 20  
set n1 [$ns node]  
$n1 set X_ 401  
$n1 set Y_ 400  
$n1 set Z_ 0.0  
$ns initial_node_pos $n1 20  
set n2 [$ns node]  
$n2 set X_ 601  
$n2 set Y_ 400  
$n2 set Z_ 0.0  
$ns initial_node_pos $n2 20  
set n3 [$ns node]  
$n3 set X_ 801  
$n3 set Y_ 400  
$n3 set Z_ 0.0  
$ns initial_node_pos $n3 20  
set n4 [$ns node]
```

```
$n4 set X_ 1001
$n4 set Y_ 400
$n4 set Z_ 0.0
$ns initial_node_pos $n4 20
set n5 [$ns node]
$n5 set X_ 1201
$n5 set Y_ 400
$n5 set Z_ 0.0
$ns initial_node_pos $n5 20
set n6 [$ns node]
$n6 set X_ 1401
$n6 set Y_ 400
$n6 set Z_ 0.0
$ns initial_node_pos $n6 20
```

```
#=====
```

```
# Agents Definition
```

```
#=====
```

```
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink6 [new Agent/TCPSink]
$ns attach-agent $n1 $sink6
$ns connect $tcp0 $sink6
$tcp0 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink7 [new Agent/TCPSink]
$ns attach-agent $n2 $sink7
$ns connect $tcp1 $sink7
$tcp1 set packetSize_ 50
```

```
#Setup a TCP connection
set tcp2 [new Agent/TCP]
$ns attach-agent $n2 $tcp2
```

```
set sink8 [new Agent/TCPSink]
$ns attach-agent $n3 $sink8
$ns connect $tcp2 $sink8
$tcp2 set packetSize_ 66
```

```
#Setup a TCP connection
set tcp3 [new Agent/TCP]
$ns attach-agent $n3 $tcp3
set sink9 [new Agent/TCPSink]
$ns attach-agent $n4 $sink9
$ns connect $tcp3 $sink9
$tcp3 set packetSize_ 82
```

```
#Setup a TCP connection
set tcp4 [new Agent/TCP]
$ns attach-agent $n4 $tcp4
set sink10 [new Agent/TCPSink]
$ns attach-agent $n5 $sink10
$ns connect $tcp4 $sink10
$tcp4 set packetSize_ 98
```

```
#Setup a TCP connection
set tcp5 [new Agent/TCP]
$ns attach-agent $n5 $tcp5
set sink11 [new Agent/TCPSink]
$ns attach-agent $n6 $sink11
$ns connect $tcp5 $sink11
$tcp5 set packetSize_ 114
```

```
#=====
```

```
# Applications Definition
```

```
#=====
```

```
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $tcp0
$cbr0 set packetSize_ 34
```

```
$cbr0 set rate_ 0.1Mb
$cbr0 set random_
$ns at 0.001 "$cbr0 start"
$ns at 0.002 "$cbr0 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $tcp1
$cbr1 set packetSize_ 50
$cbr1 set rate_ 0.1Mb
$cbr1 set random_
$ns at 0.002 "$cbr1 start"
$ns at 0.003 "$cbr1 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr2 [new Application/Traffic/CBR]
$cbr2 attach-agent $tcp2
$cbr2 set packetSize_ 66
$cbr2 set rate_ 1.0Mb
$cbr2 set random_
$ns at 0.003 "$cbr2 start"
$ns at 0.004 "$cbr2 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr3 [new Application/Traffic/CBR]
$cbr3 attach-agent $tcp3
$cbr3 set packetSize_ 82
$cbr3 set rate_ 1.0Mb
$cbr3 set random_
$ns at 0.004 "$cbr3 start"
$ns at 0.005 "$cbr3 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr4 [new Application/Traffic/CBR]
$cbr4 attach-agent $tcp4
$cbr4 set packetSize_ 98
$cbr4 set rate_ 0.1Mb
```

```

$cbr4 set random_
$ns at 0.005 "$cbr4 start"
$ns at 0.006 "$cbr4 stop"

#Setup a CBR Application over TCP connection
set cbr5 [new Application/Traffic/CBR]
$cbr5 attach-agent $tcp5
$cbr5 set packetSize_ 114
$cbr5 set rate_ 0.1Mb
$cbr5 set random_
$ns at 0.006 "$cbr5 start"
$ns at 0.007 "$cbr5 stop"

```

```

#=====
#   Termination
#=====
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
    close $tracefile
    close $namfile
    exec nam 7lampu_Z.nam &
    exit 0
}
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "\$n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run

```

4. Simulasi Topologi Jaringan 9 Node

```
#####  
# Simulation parameters setup  
#####  
Antenna/OmniAntenna set Gt_1 ;#Transmit antenna gain  
Antenna/OmniAntenna set Gr_1 ;#Receive antenna gain  
Phy/WirelessPhy set pathlossExp_ 2  
Phy/WirelessPhy set std_db_ 12  
Phy/WirelessPhy set L_ 1.0 ;#System Loss Factor  
Phy/WirelessPhy set freq_ 2.452e9 ;#channel  
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate  
Phy/WirelessPhy set Pt_ 0.06 ;#Transmit Power  
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power Threshold  
Mac/802_15_4 set dataRate_ 0.1Mb ;#Rate for Data Frames  
Mac/802_15_4 set basicRate_ 0.05Mb ;#Rate for Control  
Frames  
  
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) 50 ;# max packet in ifq  
set val(nn) 9 ;# number of mobilenodes  
set val(rp) AODV ;# routing protocol  
set val(x) 3047 ;# X dimension of topography  
set val(y) 100 ;# Y dimension of topography  
set val(stop) 10.0 ;# time of simulation end  
  
#####  
# Initialization
```

```

#=====
#Create a ns simulator
set ns [new Simulator]

#Setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)

#Open the NS trace file
set tracefile [open 9lampu_Z.tr w]
$ns trace-all $tracefile

#Open the NAM trace file
set namfile [open 9lampu_Z.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel

#=====
# Mobile node parameter setup
#=====
$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) \
    -channel $chan \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace OFF

```

```

#=====
#   Nodes Definition
#=====
#Create 9 nodes
set n0 [$ns node]
$n0 set X_ 198
$n0 set Y_ 397
$n0 set Z_ 0.0
$ns initial_node_pos $n0 20
set n1 [$ns node]
$n1 set X_ 398
$n1 set Y_ 397
$n1 set Z_ 0.0
$ns initial_node_pos $n1 20
set n2 [$ns node]
$n2 set X_ 598
$n2 set Y_ 397
$n2 set Z_ 0.0
$ns initial_node_pos $n2 20
set n3 [$ns node]
$n3 set X_ 798
$n3 set Y_ 397
$n3 set Z_ 0.0
$ns initial_node_pos $n3 20
set n4 [$ns node]
$n4 set X_ 998
$n4 set Y_ 397
$n4 set Z_ 0.0
$ns initial_node_pos $n4 20
set n5 [$ns node]
$n5 set X_ 1198
$n5 set Y_ 397
$n5 set Z_ 0.0
$ns initial_node_pos $n5 20
set n6 [$ns node]
$n6 set X_ 1398
$n6 set Y_ 397

```



```
$n6 set Z_ 0.0
$ns initial_node_pos $n6 20
set n7 [$ns node]
$n7 set X_ 1598
$n7 set Y_ 397
$n7 set Z_ 0.0
$ns initial_node_pos $n7 20
set n8 [$ns node]
$n8 set X_ 1798
$n8 set Y_ 397
$n8 set Z_ 0.0
$ns initial_node_pos $n8 20
```

```
#=====
#   Agents Definition
#=====
```

```
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink8 [new Agent/TCPSink]
$ns attach-agent $n1 $sink8
$ns connect $tcp0 $sink8
$tcp0 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink9 [new Agent/TCPSink]
$ns attach-agent $n2 $sink9
$ns connect $tcp1 $sink9
$tcp1 set packetSize_ 50
```

```
#Setup a TCP connection
set tcp2 [new Agent/TCP]
$ns attach-agent $n2 $tcp2
set sink10 [new Agent/TCPSink]
$ns attach-agent $n3 $sink10
```

```
$ns connect $tcp2 $sink10
$tcp2 set packetSize_ 66
```

```
#Setup a TCP connection
set tcp3 [new Agent/TCP]
$ns attach-agent $n3 $tcp3
set sink11 [new Agent/TCPSink]
$ns attach-agent $n4 $sink11
$ns connect $tcp3 $sink11
$tcp3 set packetSize_ 82
```

```
#Setup a TCP connection
set tcp4 [new Agent/TCP]
$ns attach-agent $n4 $tcp4
set sink12 [new Agent/TCPSink]
$ns attach-agent $n5 $sink12
$ns connect $tcp4 $sink12
$tcp4 set packetSize_ 98
```

```
#Setup a TCP connection
set tcp5 [new Agent/TCP]
$ns attach-agent $n5 $tcp5
set sink13 [new Agent/TCPSink]
$ns attach-agent $n6 $sink13
$ns connect $tcp5 $sink13
$tcp5 set packetSize_ 114
```

```
#Setup a TCP connection
set tcp6 [new Agent/TCP]
$ns attach-agent $n6 $tcp6
set sink14 [new Agent/TCPSink]
$ns attach-agent $n7 $sink14
$ns connect $tcp6 $sink14
$tcp6 set packetSize_ 148
```

```
#Setup a TCP connection
set tcp7 [new Agent/TCP]
```

```
$ns attach-agent $n7 $tcp7
set sink15 [new Agent/TCPSink]
$ns attach-agent $n8 $sink15
$ns connect $tcp7 $sink15
$tcp7 set packetSize_ 164
```

```
#=====
#   Applications Definition
#=====
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $tcp0
$cbr0 set packetSize_ 34
$cbr0 set rate_ 0.1Mb
$cbr0 set random_
$ns at 0.001 "$cbr0 start"
$ns at 0.002 "$cbr0 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $tcp1
$cbr1 set packetSize_ 50
$cbr1 set rate_ 0.1Mb
$cbr1 set random_
$ns at 0.002 "$cbr1 start"
$ns at 0.003 "$cbr1 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr2 [new Application/Traffic/CBR]
$cbr2 attach-agent $tcp2
$cbr2 set packetSize_ 66
$cbr2 set rate_ 0.1Mb
$cbr2 set random_
$ns at 0.003 "$cbr2 start"
$ns at 0.004 "$cbr2 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr3 [new Application/Traffic/CBR]
$cbr3 attach-agent $tcp3
$cbr3 set packetSize_ 82
$cbr3 set rate_ 0.1Mb
$cbr3 set random_
$ns at 0.004 "$cbr3 start"
$ns at 0.005 "$cbr3 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr4 [new Application/Traffic/CBR]
$cbr4 attach-agent $tcp4
$cbr4 set packetSize_ 98
$cbr4 set rate_ 0.1Mb
$cbr4 set random_
$ns at 0.005 "$cbr4 start"
$ns at 0.006 "$cbr4 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr5 [new Application/Traffic/CBR]
$cbr5 attach-agent $tcp5
$cbr5 set packetSize_ 114
$cbr5 set rate_ 0.1Mb
$cbr5 set random_
$ns at 0.006 "$cbr5 start"
$ns at 0.007 "$cbr5 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr6 [new Application/Traffic/CBR]
$cbr6 attach-agent $tcp6
$cbr6 set packetSize_ 148
$cbr6 set rate_ 0.1Mb
$cbr6 set random_
$ns at 0.007 "$cbr6 start"
$ns at 0.008 "$cbr6 stop"
```

```
#Setup a CBR Application over TCP connection
```

```

set cbr7 [new Application/Traffic/CBR]
$cbr7 attach-agent $tcp7
$cbr7 set packetSize_ 164
$cbr7 set rate_ 0.1Mb
$cbr7 set random_
$ns at 0.008 "$cbr7 start"
$ns at 0.009 "$cbr7 stop"

```

```

#=====
#   Termination
#=====
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
    close $tracefile
    close $namfile
    exec nam 9lampu_Z.nam &
    exit 0
}
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "\$n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run

```

5. Simulasi Topologi Jaringan 9 Node

```

#=====
#   Simulation parameters setup
#=====
Antenna/OmniAntenna set Gt_1           ;#Transmit antenna gain
Antenna/OmniAntenna set Gr_1           ;#Receive antenna gain
Phy/WirelessPhy set pathlossExp_ 2

```

```

Phy/WirelessPhy set std_db_ 12
Phy/WirelessPhy set L_ 1.0           ;#System Loss Factor
Phy/WirelessPhy set freq_ 2.452e9    ;#channel
Phy/WirelessPhy set bandwidth_ 0.25Mb ;#Data Rate
Phy/WirelessPhy set Pt_ 0.06         ;#Transmit Power
Phy/WirelessPhy set RXThresh_ 2.4164e-11 ;#Receive Power
Threshold
Mac/802_15_4 set dataRate_ 0.1Mb     ;#Rate for Data Frames
Mac/802_15_4 set basicRate_ 0.05Mb   ;#Rate for Control
Frames

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) 50                     ;# max packet in ifq
set val(nn) 17                          ;# number of mobilenodes
set val(rp) AODV                        ;# routing protocol
set val(x) 2174                          ;# X dimension of topography
set val(y) 1000                          ;# Y dimension of topography
set val(stop) 10.0                      ;# time of simulation end

#=====
# Initialization
#=====
#Create a ns simulator
set ns [new Simulator]

#Setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)

#Open the NS trace file
set tracefile [open lampu_2K.tr w]
$ns trace-all $tracefile

```

```

#Open the NAM trace file
set namfile [open lampu_2K.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel

```

```

#=====
#   Mobile node parameter setup
#=====
$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) \
    -channel     $chan \
    -topoInstance $topo \
    -agentTrace  ON \
    -routerTrace ON \
    -macTrace    ON \
    -movementTrace OFF

```

```

#=====
#   Nodes Definition
#=====
#Create 17 nodes
set n0 [$ns node]
$ns set X_ 200
$ns set Y_ 299
$ns set Z_ 0.0
$ns initial_node_pos $n0 20
set n1 [$ns node]
$ns set X_ 400
$ns set Y_ 299
$ns set Z_ 0.0
$ns initial_node_pos $n1 20

```

```
set n2 [$ns node]
$n2 set X_ 600
$n2 set Y_ 299
$n2 set Z_ 0.0
$ns initial_node_pos $n2 20
set n3 [$ns node]
$n3 set X_ 800
$n3 set Y_ 299
$n3 set Z_ 0.0
$ns initial_node_pos $n3 20
set n4 [$ns node]
$n4 set X_ 1000
$n4 set Y_ 299
$n4 set Z_ 0.0
$ns initial_node_pos $n4 20
set n5 [$ns node]
$n5 set X_ 1200
$n5 set Y_ 299
$n5 set Z_ 0.0
$ns initial_node_pos $n5 20
set n6 [$ns node]
$n6 set X_ 1400
$n6 set Y_ 299
$n6 set Z_ 0.0
$ns initial_node_pos $n6 20
set n7 [$ns node]
$n7 set X_ 1600
$n7 set Y_ 299
$n7 set Z_ 0.0
$ns initial_node_pos $n7 20
set n8 [$ns node]
$n8 set X_ 201
$n8 set Y_ 197
$n8 set Z_ 0.0
$ns initial_node_pos $n8 20
set n9 [$ns node]
$n9 set X_ 401
$n9 set Y_ 197
$n9 set Z_ 0.0
```



```
$ns initial_node_pos $n9 20
set n10 [$ns node]
$ns10 set X_ 601
$ns10 set Y_ 197
$ns10 set Z_ 0.0
$ns initial_node_pos $n10 20
set n11 [$ns node]
$ns11 set X_ 801
$ns11 set Y_ 197
$ns11 set Z_ 0.0
$ns initial_node_pos $n11 20
set n12 [$ns node]
$ns12 set X_ 1001
$ns12 set Y_ 197
$ns12 set Z_ 0.0
$ns initial_node_pos $n12 20
set n13 [$ns node]
$ns13 set X_ 1201
$ns13 set Y_ 197
$ns13 set Z_ 0.0
$ns initial_node_pos $n13 20
set n14 [$ns node]
$ns14 set X_ 1401
$ns14 set Y_ 197
$ns14 set Z_ 0.0
$ns initial_node_pos $n14 20
set n15 [$ns node]
$ns15 set X_ 1601
$ns15 set Y_ 197
$ns15 set Z_ 0.0
$ns initial_node_pos $n15 20
set n16 [$ns node]
$ns16 set X_ 1701
$ns16 set Y_ 244
$ns16 set Z_ 0.0
$ns initial_node_pos $n16 20
```

```
#=====
# Agents Definition
```

#=====

```
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink4 [new Agent/TCPSink]
$ns attach-agent $n1 $sink4
$ns connect $tcp0 $sink4
$tcp0 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n8 $tcp1
set sink6 [new Agent/TCPSink]
$ns attach-agent $n9 $sink6
$ns connect $tcp1 $sink6
$tcp1 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp2 [new Agent/TCP]
$ns attach-agent $n0 $tcp2
set sink7 [new Agent/TCPSink]
$ns attach-agent $n9 $sink7
$ns connect $tcp2 $sink7
$tcp2 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp3 [new Agent/TCP]
$ns attach-agent $n8 $tcp3
set sink5 [new Agent/TCPSink]
$ns attach-agent $n1 $sink5
$ns connect $tcp3 $sink5
$tcp3 set packetSize_ 34
```

```
#Setup a TCP connection
set tcp8 [new Agent/TCP]
$ns attach-agent $n1 $tcp8
set sink12 [new Agent/TCPSink]
$ns attach-agent $n2 $sink12
$ns connect $tcp8 $sink12
```

```
$tcp8 set packetSize_ 66
```

```
#Setup a TCP connection  
set tcp9 [new Agent/TCP]  
$ns attach-agent $n1 $tcp9  
set sink15 [new Agent/TCPSink]  
$ns attach-agent $n10 $sink15  
$ns connect $tcp9 $sink15  
$tcp9 set packetSize_ 66
```

```
#Setup a TCP connection  
set tcp10 [new Agent/TCP]  
$ns attach-agent $n9 $tcp10  
set sink14 [new Agent/TCPSink]  
$ns attach-agent $n10 $sink14  
$ns connect $tcp10 $sink14  
$tcp10 set packetSize_ 66
```

```
#Setup a TCP connection  
set tcp11 [new Agent/TCP]  
$ns attach-agent $n9 $tcp11  
set sink13 [new Agent/TCPSink]  
$ns attach-agent $n2 $sink13  
$ns connect $tcp11 $sink13  
$tcp11 set packetSize_ 66
```

```
#Setup a TCP connection  
set tcp16 [new Agent/TCP]  
$ns attach-agent $n2 $tcp16  
set sink20 [new Agent/TCPSink]  
$ns attach-agent $n3 $sink20  
$ns connect $tcp16 $sink20  
$tcp16 set packetSize_ 98
```

```
#Setup a TCP connection  
set tcp17 [new Agent/TCP]  
$ns attach-agent $n2 $tcp17  
set sink23 [new Agent/TCPSink]  
$ns attach-agent $n11 $sink23
```

```
$ns connect $tcp17 $sink23
$tcp17 set packetSize_ 98
```

```
#Setup a TCP connection
set tcp18 [new Agent/TCP]
$ns attach-agent $n10 $tcp18
set sink22 [new Agent/TCPSink]
$ns attach-agent $n11 $sink22
$ns connect $tcp18 $sink22
$tcp18 set packetSize_ 98
```

```
#Setup a TCP connection
set tcp19 [new Agent/TCP]
$ns attach-agent $n10 $tcp19
set sink21 [new Agent/TCPSink]
$ns attach-agent $n3 $sink21
$ns connect $tcp19 $sink21
$tcp19 set packetSize_ 98
```

```
#Setup a TCP connection
set tcp24 [new Agent/TCP]
$ns attach-agent $n3 $tcp24
set sink28 [new Agent/TCPSink]
$ns attach-agent $n4 $sink28
$ns connect $tcp24 $sink28
$tcp24 set packetSize_ 148
```

```
#Setup a TCP connection
set tcp25 [new Agent/TCP]
$ns attach-agent $n3 $tcp25
set sink31 [new Agent/TCPSink]
$ns attach-agent $n12 $sink31
$ns connect $tcp25 $sink31
$tcp25 set packetSize_ 148
```

```
#Setup a TCP connection
set tcp26 [new Agent/TCP]
$ns attach-agent $n11 $tcp26
set sink30 [new Agent/TCPSink]
```

```
$ns attach-agent $n12 $sink30
$ns connect $tcp26 $sink30
$tcp26 set packetSize_ 148
```

```
#Setup a TCP connection
set tcp27 [new Agent/TCP]
$ns attach-agent $n11 $tcp27
set sink29 [new Agent/TCPSink]
$ns attach-agent $n4 $sink29
$ns connect $tcp27 $sink29
$tcp27 set packetSize_ 148
```

```
#Setup a TCP connection
set tcp32 [new Agent/TCP]
$ns attach-agent $n4 $tcp32
set sink36 [new Agent/TCPSink]
$ns attach-agent $n5 $sink36
$ns connect $tcp32 $sink36
$tcp32 set packetSize_ 180
```

```
#Setup a TCP connection
set tcp33 [new Agent/TCP]
$ns attach-agent $n4 $tcp33
set sink39 [new Agent/TCPSink]
$ns attach-agent $n13 $sink39
$ns connect $tcp33 $sink39
$tcp33 set packetSize_ 180
```

```
#Setup a TCP connection
set tcp34 [new Agent/TCP]
$ns attach-agent $n12 $tcp34
set sink38 [new Agent/TCPSink]
$ns attach-agent $n13 $sink38
$ns connect $tcp34 $sink38
$tcp34 set packetSize_ 180
```

```
#Setup a TCP connection
set tcp35 [new Agent/TCP]
$ns attach-agent $n12 $tcp35
```

```
set sink37 [new Agent/TCPSink]
$ns attach-agent $n5 $sink37
$ns connect $tcp35 $sink37
$tcp35 set packetSize_ 180
```

```
#Setup a TCP connection
set tcp40 [new Agent/TCP]
$ns attach-agent $n5 $tcp40
set sink44 [new Agent/TCPSink]
$ns attach-agent $n6 $sink44
$ns connect $tcp40 $sink44
$tcp40 set packetSize_ 212
```

```
#Setup a TCP connection
set tcp41 [new Agent/TCP]
$ns attach-agent $n5 $tcp41
set sink47 [new Agent/TCPSink]
$ns attach-agent $n14 $sink47
$ns connect $tcp41 $sink47
$tcp41 set packetSize_ 212
```

```
#Setup a TCP connection
set tcp42 [new Agent/TCP]
$ns attach-agent $n13 $tcp42
set sink46 [new Agent/TCPSink]
$ns attach-agent $n14 $sink46
$ns connect $tcp42 $sink46
$tcp42 set packetSize_ 212
```

```
#Setup a TCP connection
set tcp43 [new Agent/TCP]
$ns attach-agent $n13 $tcp43
set sink45 [new Agent/TCPSink]
$ns attach-agent $n6 $sink45
$ns connect $tcp43 $sink45
$tcp43 set packetSize_ 212
```

```
#Setup a TCP connection
set tcp48 [new Agent/TCP]
```

```
$ns attach-agent $n6 $tcp48
set sink52 [new Agent/TCPSink]
$ns attach-agent $n7 $sink52
$ns connect $tcp48 $sink52
$tcp48 set packetSize_ 260
```

```
#Setup a TCP connection
set tcp49 [new Agent/TCP]
$ns attach-agent $n6 $tcp49
set sink55 [new Agent/TCPSink]
$ns attach-agent $n15 $sink55
$ns connect $tcp49 $sink55
$tcp49 set packetSize_ 260
```

```
#Setup a TCP connection
set tcp50 [new Agent/TCP]
$ns attach-agent $n14 $tcp50
set sink54 [new Agent/TCPSink]
$ns attach-agent $n15 $sink54
$ns connect $tcp50 $sink54
$tcp50 set packetSize_ 260
```

```
#Setup a TCP connection
set tcp51 [new Agent/TCP]
$ns attach-agent $n14 $tcp51
set sink53 [new Agent/TCPSink]
$ns attach-agent $n7 $sink53
$ns connect $tcp51 $sink53
$tcp51 set packetSize_ 260
```

```
#Setup a TCP connection
set tcp56 [new Agent/TCP]
$ns attach-agent $n7 $tcp56
set sink58 [new Agent/TCPSink]
$ns attach-agent $n16 $sink58
$ns connect $tcp56 $sink58
$tcp56 set packetSize_ 292
```

```
#Setup a TCP connection
```

```
set tcp57 [new Agent/TCP]
$ns attach-agent $n15 $tcp57
set sink59 [new Agent/TCPSink]
$ns attach-agent $n16 $sink59
$ns connect $tcp57 $sink59
$tcp57 set packetSize_ 292
```

```
#=====
# Applications Definition
#=====
```

```
#Setup a CBR Application over TCP connection
set cbr0 [new Application/Traffic/CBR]
$scbr0 attach-agent $tcp0
$scbr0 set packetSize_ 34
$scbr0 set rate_ 0.1Mb
$scbr0 set random_
$ns at 0.001 "$scbr0 start"
$ns at 0.002 "$scbr0 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr1 [new Application/Traffic/CBR]
$scbr1 attach-agent $tcp1
$scbr1 set packetSize_ 34
$scbr1 set rate_ 0.1Mb
$scbr1 set random_ null
$ns at 0.001 "$scbr1 start"
$ns at 0.002 "$scbr1 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr2 [new Application/Traffic/CBR]
$scbr2 attach-agent $tcp8
$scbr2 set packetSize_ 66
$scbr2 set rate_ 0.1Mb
$scbr2 set random_ null
$ns at 0.002 "$scbr2 start"
$ns at 0.003 "$scbr2 stop"
```

```
#Setup a CBR Application over TCP connection
```



```
set cbr3 [new Application/Traffic/CBR]
$scr3 attach-agent $tcp9
$scr3 set packetSize_ 66
$scr3 set rate_ 0.1Mb
$scr3 set random_ null
$ns at 0.002 "$scr3 start"
$ns at 0.003 "$scr3 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr4 [new Application/Traffic/CBR]
$scr4 attach-agent $tcp10
$scr4 set packetSize_ 66
$scr4 set rate_ 0.1Mb
$scr4 set random_ null
$ns at 0.002 "$scr4 start"
$ns at 0.003 "$scr4 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr5 [new Application/Traffic/CBR]
$scr5 attach-agent $tcp11
$scr5 set packetSize_ 66
$scr5 set rate_ 0.1Mb
$scr5 set random_ null
$ns at 0.002 "$scr5 start"
$ns at 0.003 "$scr5 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr6 [new Application/Traffic/CBR]
$scr6 attach-agent $tcp16
$scr6 set packetSize_ 98
$scr6 set rate_ 0.1Mb
$scr6 set random_ null
$ns at 0.003 "$scr6 start"
$ns at 0.004 "$scr6 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr7 [new Application/Traffic/CBR]
$scr7 attach-agent $tcp17
$scr7 set packetSize_ 98
```

```
$cbr7 set rate_ 0.1Mb  
$cbr7 set random_ null  
$ns at 0.003 "$cbr7 start"  
$ns at 0.004 "$cbr7 stop"
```

```
#Setup a CBR Application over TCP connection  
set cbr8 [new Application/Traffic/CBR]  
$cbr8 attach-agent $tcp18  
$cbr8 set packetSize_ 98  
$cbr8 set rate_ 0.1Mb  
$cbr8 set random_ null  
$ns at 0.003 "$cbr8 start"  
$ns at 0.004 "$cbr8 stop"
```

```
#Setup a CBR Application over TCP connection  
set cbr9 [new Application/Traffic/CBR]  
$cbr9 attach-agent $tcp19  
$cbr9 set packetSize_ 98  
$cbr9 set rate_ 0.1Mb  
$cbr9 set random_ null  
$ns at 0.003 "$cbr9 start"  
$ns at 0.004 "$cbr9 stop"
```

```
#Setup a CBR Application over TCP connection  
set cbr10 [new Application/Traffic/CBR]  
$cbr10 attach-agent $tcp24  
$cbr10 set packetSize_ 148  
$cbr10 set rate_ 0.1Mb  
$cbr10 set random_  
$ns at 0.004 "$cbr10 start"  
$ns at 0.005 "$cbr10 stop"
```

```
#Setup a CBR Application over TCP connection  
set cbr11 [new Application/Traffic/CBR]  
$cbr11 attach-agent $tcp25  
$cbr11 set packetSize_ 148  
$cbr11 set rate_ 0.1Mb  
$cbr11 set random_ null  
$ns at 0.004 "$cbr11 start"
```

\$ns at 0.005 "\$cbr11 stop"

```
#Setup a CBR Application over TCP connection
set cbr12 [new Application/Traffic/CBR]
$cbr12 attach-agent $tcp26
$cbr12 set packetSize_ 148
$cbr12 set rate_ 0.1Mb
$cbr12 set random_ null
$ns at 0.004 "$cbr12 start"
$ns at 0.005 "$cbr12 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr13 [new Application/Traffic/CBR]
$cbr13 attach-agent $tcp27
$cbr13 set packetSize_ 148
$cbr13 set rate_ 0.1Mb
$cbr13 set random_
$ns at 0.004 "$cbr13 start"
$ns at 0.005 "$cbr13 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr14 [new Application/Traffic/CBR]
$cbr14 attach-agent $tcp32
$cbr14 set packetSize_ 180
$cbr14 set rate_ 0.1Mb
$cbr14 set random_
$ns at 0.005 "$cbr14 start"
$ns at 0.006 "$cbr14 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr15 [new Application/Traffic/CBR]
$cbr15 attach-agent $tcp33
$cbr15 set packetSize_ 180
$cbr15 set rate_ 0.1Mb
$cbr15 set random_
$ns at 0.005 "$cbr15 start"
$ns at 0.006 "$cbr15 stop"
```

```
#Setup a CBR Application over TCP connection
```

```
set cbr16 [new Application/Traffic/CBR]
$cbr16 attach-agent $tcp34
$cbr16 set packetSize_ 180
$cbr16 set rate_ 0.1Mb
$cbr16 set random_
$ns at 0.005 "$cbr16 start"
$ns at 0.006 "$cbr16 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr17 [new Application/Traffic/CBR]
$cbr17 attach-agent $tcp35
$cbr17 set packetSize_ 180
$cbr17 set rate_ 0.1Mb
$cbr17 set random_
$ns at 0.005 "$cbr17 start"
$ns at 0.006 "$cbr17 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr18 [new Application/Traffic/CBR]
$cbr18 attach-agent $tcp40
$cbr18 set packetSize_ 212
$cbr18 set rate_ 0.1Mb
$cbr18 set random_
$ns at 0.006 "$cbr18 start"
$ns at 0.007 "$cbr18 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr19 [new Application/Traffic/CBR]
$cbr19 attach-agent $tcp41
$cbr19 set packetSize_ 212
$cbr19 set rate_ 0.1Mb
$cbr19 set random_
$ns at 0.006 "$cbr19 start"
$ns at 0.007 "$cbr19 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr20 [new Application/Traffic/CBR]
$cbr20 attach-agent $tcp42
$cbr20 set packetSize_ 212
```

```
$cbr20 set rate_ 0.1Mb
$cbr20 set random_
$ns at 0.006 "$cbr20 start"
$ns at 0.007 "$cbr20 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr21 [new Application/Traffic/CBR]
$cbr21 attach-agent $tcp43
$cbr21 set packetSize_ 212
$cbr21 set rate_ 0.1Mb
$cbr21 set random_
$ns at 0.006 "$cbr21 start"
$ns at 0.007 "$cbr21 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr22 [new Application/Traffic/CBR]
$cbr22 attach-agent $tcp48
$cbr22 set packetSize_ 260
$cbr22 set rate_ 0.1Mb
$cbr22 set random_ null
$ns at 0.007 "$cbr22 start"
$ns at 0.008 "$cbr22 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr23 [new Application/Traffic/CBR]
$cbr23 attach-agent $tcp49
$cbr23 set packetSize_ 260
$cbr23 set rate_ 0.1Mb
$cbr23 set random_ null
$ns at 0.007 "$cbr23 start"
$ns at 0.008 "$cbr23 stop"
```

```
#Setup a CBR Application over TCP connection
set cbr24 [new Application/Traffic/CBR]
$cbr24 attach-agent $tcp50
$cbr24 set packetSize_ 260
$cbr24 set rate_ 0.1Mb
$cbr24 set random_ null
$ns at 0.007 "$cbr24 start"
```

```
$ns at 0.008 "$cbr24 stop"
```

```
#Setup a CBR Application over TCP connection
```

```
set cbr25 [new Application/Traffic/CBR]
```

```
$cbr25 attach-agent $tcp51
```

```
$cbr25 set packetSize_ 260
```

```
$cbr25 set rate_ 0.1Mb
```

```
$cbr25 set random_ null
```

```
$ns at 0.007 "$cbr25 start"
```

```
$ns at 0.008 "$cbr25 stop"
```

```
#Setup a CBR Application over TCP connection
```

```
set cbr26 [new Application/Traffic/CBR]
```

```
$cbr26 attach-agent $tcp56
```

```
$cbr26 set packetSize_ 292
```

```
$cbr26 set rate_ 0.1Mb
```

```
$cbr26 set random_ null
```

```
$ns at 0.008 "$cbr26 start"
```

```
$ns at 0.009 "$cbr26 stop"
```

```
#Setup a CBR Application over TCP connection
```

```
set cbr27 [new Application/Traffic/CBR]
```

```
$cbr27 attach-agent $tcp57
```

```
$cbr27 set packetSize_ 292
```

```
$cbr27 set rate_ 0.1Mb
```

```
$cbr27 set random_ null
```

```
$ns at 0.008 "$cbr27 start"
```

```
$ns at 0.009 "$cbr27 stop"
```

```
#=====
```

```
# Termination
```

```
#=====
```

```
#Define a 'finish' procedure
```

```
proc finish {} {
```

```
    global ns tracefile namfile
```

```
    $ns flush-trace
```

```
    close $tracefile
```

```
    close $namfile
```

```
exec nam lampu_2K.nam &
exit 0
}
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "\n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run
```

[Halaman ini sengaja dikosongkan]

BIOGRAFI PENULIS



Muhammad Ilham Pratomo merupakan salah satu mahasiswa di Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Penulis lahir di Bandung pada tanggal 18 Juni 1996. Fakultas yang diambilnya adalah Fakultas Teknologi Elektro (FTE) dengan jurusan Departemen Teknik Elektro. Pada peminatan bidang studi tertarik untuk memilih bidang studi Telekomunikasi Multimedia (TMM). Memiliki hobi *traveling* dan mencari kuliner, bercita-cita menjadi pengusaha di bidang teknologi agar dapat menjadikan negara Indonesia menjadi bangsa yang terdepan dalam bidang teknologi informasi dan komunikasi.