

25338/H/06



**EVALUASI KINERJA PROTOKOL PERUTEAN
PROAKTIF PADA JARINGAN NIRKABEL AD HOC
MULTIHOP**

Oleh :

**AMRI
2203.203.005**

RTE
621.384
AMR
E-1
2009



PERPUSTAKAAN ITS	
Tgl. Terima	10-3-06
Terima Dari	H
No. Agenda Prp.	229006

**PROGRAM STUDI MAGISTER
BIDANG KEAHLIAN TELEKOMUNIKASI MULTIMEDIA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2005**

Evaluasi Kinerja Protokol Perutean Proaktif Pada Jaringan Nirkabel Ad Hoc Multihop

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (MT)
di
Institut Teknologi Sepuluh Nopember

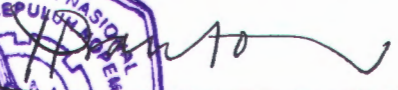
oleh:
Amri
Nrp. 2203 203 005

Tanggal Ujian: 23 Juli 2005
Periode Wisuda : September 2005


Disetujui oleh Tim Penguji Tesis:

1. Dr.Ir. Gamantyo Hendrantoro, M.Eng. (Pembimbing I)
NIP. 132 048 147
2. Eko Setijadi, ST, MT
NIP. 132 304 283
3. Dr. Ir. Wirawan, DEA
NIP. 131 842 501
4. Dr. Ir. Achmad Affandi, DEA
NIP. 131 879 394
5. Ir. Suwadi, MT
NIP. 132 053 513
6. Ir. Titiek Suryani, MT
NIP. 131 841 930
7. Ir. Achmad Ansori, DEA
NIP. 131 855 878
8. Dr. Ir. Titon Dutono, M.Eng
NIP. 131 651 258

Direktur Program Pascasarjana



Prof. Ir. Happy Ratna S, MSc., PhD
NIP. 130 541 829



EVALUASI KINERJA PROTOKOL PERUTEAN PADA JARINGAN NIRKABEL AD HOC MULTIHOP

Nama Mahasiswa : Amri
NRP : 2203 203 005
Pembimbing : Dr. Ir. Gamantyo Hendranto, M.Eng.
Co-Pembimbing : Eko Setijadi, ST, MT.

ABSTRAK

Jaringan *wireless ad hoc* adalah sekumpulan node bergerak pada *wireless* secara dinamis membentuk suatu jaringan temporer tanpa menggunakan infrastruktur jaringan yang ada atau administrasi terpusat. Sejumlah *route* protokol proaktif yang diteliti seperti OLSR (*Optimized Link State Route*) dan FSR (*Fisheye State Route*) sudah banyak diimplementasikan.

Pada penelitian ini dilakukan percobaan menggunakan simulator level-paket pada 100 node bergerak secara acak pada lokasi 1000 x 1000 meter persegi, untuk mengevaluasi dan membandingkan unjuk kerja kedua protocol rute proaktif pada jaringan *wireless ad hoc multi-hop*. Hasil penelitian pengiriman dan perawatan rute merupakan faktor terpenting yang mempengaruhi pengukuran unjuk kerja khususnya dalam hal laju keberhasilan penyampaian paket pada beban kerja jaringan tinggi dan peningkatan kecepatan node. Protokol dengan dasar kerja tabel, *table-driven*, OLSR (*Optimized Link State Route Protocol*), memiliki unjuk kerja paling baik dibandingkan dengan (*Fisheye State Rouring*) FSR. Unjuk kerja rute protocol ini dianalisa dengan menggunakan waktu jeda dan kemampuan untuk mencari rute sehingga didapatkan *throughput* terbaik, waktu tunda (*delay*) tersingkat, perubahan kecepatan gerakan node dan ukuran area tertentu.

Untuk memperoleh hasil yang diharapkan, dengan menggunakan model simulasi *Network Simulator (NS-2)* Allinone versi 2.28, untuk ketepatan model lapisan MAC dan sifat lapisan fisik baku *wireless LAN IEEE 802.11* yang digunakan pada jaringan *wireless ad hoc multi-hop*.

Perbandingan unjuk kerja protocol rute proaktif akan mendapat penilaian yang terbaik kualitas protocol rute yang diteliti berdasarkan keberhasilan penyampaian paket, *throughput* dan End-to-End delay adalah: *Optimized Link State Route* (OLSR) dengan rata-rata *throughput* mencapai 99,5 %, juga merupakan protokol dengan unjuk kerja efisiensi protokol paling baik sedangkan *Fisheye State Route* (FSR) merupakan protokol rute yang memiliki unjuk kerja tunda waktu penyampaian paket dari node sumber ke node tujuan terpendek.

Kata kunci : jaringan *ad hoc*, *wireless*, proaktif, rute, *ns-2*

PERFORMANCE EVALUATION OF MULTI-HOP WIRELESS AD HOC NETWORK PROACTIVE ROUTING PROTOCOLS

Name : Amri
NRP : 2203 203 005
Supervisor : Dr. Ir. Gamantyo Hendrantoro, M.Eng.
Co-Supervisor : Eko Setijadi, ST, MT.

ABSTRACT

An ad hoc wireless network is a collection of wireless mobile nodes dynamically forming a temporary network without the use of network infrastructure or centralized administration. A number of proactive routing protocols like Optimized Link State Routing (OLSR) and Fisheye State Routing (FSR) have been implemented.

In this project an attempt has been made to compare the performance each protocol using a detailed packet-level simulator for 100 mobile nodes in location 1000x1000 square meter, wireless ad hoc network. Results have shown that quick management of route maintenance is an important factor that effects all the performance measures, especially the successful delivery rate at high workloads and increasing speeds. Protocol based on table-driven, OLSR (Optimized Link State Routing), have poor performance. The source-initiated protocol like have better performance(Fisheye State Routing) FSR. Performances comparing analyzed with network workload variation, and mobile nodes speed variation on a fixed square area.

Our simulation experiment used the *ns-2* network simulator Version Allinone 2.28, to accurately model the MAC layer and physical layer behavior of the 802.11 wireless LAN standard for ad hoc wireless network multihop.

Performance comparison for proactive routing protocol will be worked on routing protocol quality based on packet delivery, throughput and end-to-end delay. OLSR that reach throughput rate 99.5% is a protocol with the bast performance efeciency, mean while FSR is a routing protocol with the shortest end-to-end delay.

Keywords : ad hoc network, wireless, routing, Proactive, *ns-2*

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT atas segala rahmat, taufik dan hidayah yang telah dilimpahkanNya hingga tersusunnya tesis *Evaluasi Kinerja Protokol Perutean Proaktif pada Jaringan Nirkabel Ad Hoc Multihop*, sebagai salah satu persyaratan untuk memperoleh gelar Magister Teknik.

Penghargaan dan terima kasih yang tak terhingga saya sampaikan kepada:

1. Bapak-bapak dosen pembimbing: Dr. Ir. Gamantyo Hendranto, M.Eng.; Eko Setijadi, ST, MT. yang dengan ketekunan dan kesabarannya telah membimbing dan mendorong penulis untuk menyelesaikan tesis ini.
2. Bapak Koordinator S2 Teknik Telekomunikasi, Bapak-Ibu dosen pengajar yang sangat banyak memberikan masukan dan bimbingan terhadap penelitian dan penulisan tesis ini. Serta segenap staf pengajar dan karyawan Jurusan Teknik Elektro ITS Surabaya.
3. Istri dan anak-anak tercinta atas dukungan dan segala bantuan yang telah diberikan. Serta segenap karib, saudara dan kawan-kawan semasa kuliah yang tak mungkin saya sebutkan satu persatu.
4. Ayah dan Almarhum Ibunda tercinta yang telah memberikan doa dan dorongan selama menjalani perkuliahan hingga terselesainya tesis ini

Akhirnya, tiada gading yang tak retak, saya selalu menunggu kritik, saran, dan masukan yang membangun untuk kesempurnaan penelitian yang telah saya lakukan. Semoga penelitian ini bermanfaat bagi perkembangan ilmu pengetahuan.

Jurusan Teknik Elektro FTI ITS
Surabaya

Surabaya, 23 Juli 2005

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
ABSTRAK	iii
KATA PENGANTAR	iv
DAFTAR ISI	v
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
DAFTAR SINGKATAN / ISTILAH	xii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan dan Batasan Masalah	3
1.3. Tinjauan Penelian	4
1.4. Manfaat Penelitian	5
1.5. Sistematika Penelitian	5
BAB II TINJAUAN PUSTAKA	
2.1. Jaringan Wireless	6
2.2. Wireless Local Area Network (WLAN)	7
2.2.1. Arsitektur <i>Wireless Local Area Network</i>	9
2.2.1.1. Komponen-Komponen dalam WLAN	9
2.2.1.2. Topologi Jaringan WLAN	12
2.2.1. <i>Layer-Layer</i> dalam WLAN	13
2.3. Media Pengaksesan	14

2.2.2. Medium Access Control Protocol	14
2.2.3. Lapisan Fisik 802.11	17
2.4. Protokol Rute	20
2.4.1. Studi Literatur	20
2.4.1.1. Penelitian Terdahulu	21
2.5. Rute Protokol Proaktif	24
2.5.1. OLSR (Optimized Link State Routing)	24
2.5.1.1. Fisheye State Routing (FSR)	25
2.5.2. Prinsip Dasar Protokol Rute	26
2.5.2.1. Distance Vector Routing	26
2.5.2.2. Link State Routing	26
2.5.2.3. Source Routing	27
2.5.3. Sifat-sifat Protokol Rute Jaringan Radio ad-hoc yang diharapkan	27
2.5.3.1. Quality of Service	28
2.5.3.2. Delay	29
2.4.3.3. Throughput	31
BAB III METODOLOGI PENELITIAN	
3.1. Model Simulasi	32
3.1.1. Network Simulator	32
3.1.2. Pemodelan Lapisan Fisik	34
3.1.2.1. Implementasi Protokol MAC 802.11	35
3.1.3. Pergerakan Node	36

3.2. Blok Diagram Simulasi	38
3.2.1. Pola Gerakan Node dan Pola Trafik	49
3.2.2. Routing Protokol Proactive yang Dievaluasi	43
3.2.2.1. OLSR (Optimized Link State Routing)	43
3.2.2.2. FSR (Fisheye State Routing)	45
3.2.3. Evaluasi Perbandingan Unjuk Kerja Routing Protokol Jaringan Wireless Ad Hoc	46
3.2.3.1. Keberhasilan Penyampaian Paket	47
3.2.3.3. Throughput	47
3.2.4. Faktor-faktor Simulasi	48
3.2.4.1. Model Gerakan	49
3.2.5. Set-up Simulasi	49
BAB IV HASIL DAN PEMBAHASAN	
4.1. Gerakan Node	51
4.1.1. Keberhasilan Penyampaian Paket	52
4.1.2. Tunda Waktu antara Node Sumber ke Node Tujuan	54
4.1.3. Throughput antara Node Sumber ke Node Tujuan	56
4.1.4. Rangkiman Akibat Gerakan Node	57
4.2. Akibat Kenaikan Beban Kerja	58
4.2.1. Keberhasilan Penyampaian Paket	59
4.2.2. Waktu Tunda antara Node Sumber ke Node Tujuan	62
4.2.3. Throughput antara Node Sumber ke node Tujuan	64
4.3. Perbandingan dari Protokol-protokol Rute	66

BAB V PENUTUP

5.1. Kesimpulan 69

5.2. Saran 70

DAFTAR PUSTAKA 72

LAMPIRAN-LAMPIRAN

I. Seting Network Simulator 74

II. Tabulasi Data Penelitian Komparasi Wireless Ad Hoc Routing Protocol 86



DAFTAR GAMBAR

Gambar 1.1.	Jaringan radio ad-hoc sederhana	3
Gambar 2.1.	Access Point dan Wireless LAN Interface	10
Gambar 2.2.	Topologi jaringan WLAN	12
Gambar 2.3.	Posisi layer WLAN dalam model OSI	14
Gambar 2.4.	Sinkronisasi 4 langkah CSMA/CA	15
Gambar 2.5.	Problem stasiun tersembunyi/hidden station	16
Gambar 2.6.	Model referensi IEEE 802.11	17
Gambar 2.7.	Format frame PLCP secara umum	18
Gambar 2.8.	File PLCP untuk FHSS	18
Gambar 2.9.	Frame PLCP untuk DSSS	18
Gambar 2.10.	Akurasi informasi pada FSR	25
Gambar 3.1.	Bagan kerja simulator <i>ns-2</i>	33
Gambar 3.2.	Node bergerak (mobile node) <i>ns-2</i>	38
Gambar 3.3.	Blok diagram proses simulasi	49
Gambar 3.4.	Gambaran topologi gerakan Mobile Node dan koneksi komunikasi pada proses simulasi di mesin <i>ns-2</i>	41
Gambar 3.5.	Flooding menggunakan mekanisme Multipoint Relay (MPR)	45

Gambar 4.1.	Keberhasilan Penyampaian Paket	53
Gambar 4.2.	Waktu tunda antara node sumber ke node tujuan rata-rata fungsi waktu jeda	55
Gambar 4.3.	Throughput rata-rata antara node sumber ke node tujuan	57
Gambar 4.4.	Keberhasilan penyampaian paket pada protokol rute OLSR	60
Gambar 4.5.	Keberhasilan penyampaian paket rute protokol FSR	61
Gambar 4.6.	Waktu tunda End-to-end rute protokol OLSR	62
Gambar 4.7.	Waktu tunda End-to-end rute protokol FSR	63
Gambar 4.8.	Throughput rata-rata protokol OLSR	65
Gambar 4.9.	Throughput rata-rata protokol FSR	65

DAFTAR TABEL

Tabel 3.1.	Definisi beban kerja untuk bermacam pola trafik	42
Tabel 3.2.	Konstanta yang digunakan pada Simulasi OLSR	43
Tabel 3.3.	Konstanta yang digunakan pada Simulasi FSR	45
Tabel 3.4.	Parameter Set Up Simulasi	50
Tabel 4.1.	Rasio keberhasilan penyampaian paket	53
Tabel 4.2.	Waktu tunda penyampaian paket	54
Tabel 4.3.	Throughput Rata-rata	56
Tabel 4.4.	Rasio keberhasilan penyampaian paket OLSR (pecahan/prosen)	60
Tabel 4.5.	Rasio keberhasilan penyampaian paket FSR (pecahan/prosen)	61
Tabel 4.6.	Waktu tunda rata-rata OLSR (detik)	62
Tabel 4.7.	Waktu tunda rata-rata FSR (detik)	63
Tabel 4.8.	Rata-rata throughput OLSR (kbps)	64
Tabel 4.9.	Rata-rata throughput FSR (kbps)	65
Tabel 4.10.	Perbandingan rute protokol proaktif	67
Tabel 4.11.	Rangkuman penilaian terhadap protokol-protokol rute	68

DAFTAR SINGKATAN / ISTILAH

OLSR	Optimized Link State Routing
FSR	Fisheye State Routing
DSR	Dynamic Source Routing
AODV	Ad Hoc On-Demand Distance Vector
ACK	Acknowledgement
ZRP	Zone Routing Protocol
CGSR	Cluster-head Gateway Switch Routing Protocol
WRP	Wireless Routing Protocol
GSR	Global State Routing
HSR	Hierarchical State Routing
QoS	Quality of Service
DoS	Denial of Service
CEDAR	Core Extraction Distributed Ad hoc Routing Algorithm
ODMRP	On-Demand Multicast Routing Protocol
GPS	Global Positioning System
MCN	Multi-hop Cellular Network
MAC	Medium Access Control
PLCP	Physical Layer Convergence Procedure
PSDU	Physical Service Data Unit
PMD	Physical Media Dependent
PLW	PSDU Length Word
PSF	PLCP Signaling Field
DSSS	Direct Sequenced Spread Spectrum
FHSS	Frequency Hop Spread Spectrum
CSMA/CD	Carrier Sense Multiple Access / Collision Detection
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
MMDS	Multi-channel Multipoint Distribution Service
LMDS	Local Multipoint Distribution Service

BAB I

PENDAHULUAN

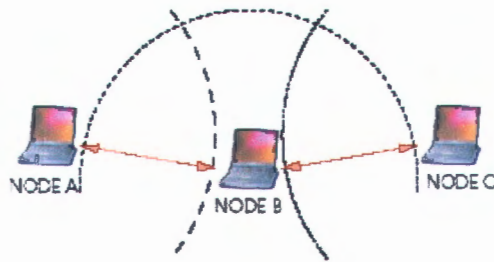
1.1. Latar Belakang

Perkembangan komunikasi radio bergerak saat ini sangat pesat, dimulai dari aktivitas pelayanan sistem komunikasi nirkabel (*Wireless*) seperti telepon, radio amatir, pengiriman berita secara tertulis, system radio microwave dan telekomunikasi generasi ke tiga yaitu yang dirancang untuk aplikasi multimedia. Penggunaan teknologi terbaru merupakan infrastruktur system informasi yang harus diterapkan sesuai dengan kebutuhan personal komunikasi yang lebih maju, khususnya system komunikasi radio nirkabel. Penggunaannya meluas ke beberapa bidang pelayanan dengan memerlukan ketepatan dan kecepatan untuk penyampaian data dari stasiun asal ke stasiun tujuan. Kemudian berkembang dari komunikasi radio paket *point to point* menjadi suatu jaringan komunikasi radio paket (terintegrasi ataupun *multihop*) maka diperlukan suatu arsitektur jaringan dan teknik *routing* yang memenuhi kebutuhan komunitas pengguna jaringan.

Pada teknik akses radio paket para pengguna bersama-sama menggunakan sebuah kanal yang tak terkoordinasi (atau dengan koordinasi minimal), hal ini sangat mudah untuk diimplementasikan, akan tetapi memiliki efisiensi spektral rendah dan kemungkinan terjadinya waktu tunda. Para pengguna menggunakan teknik pengisian (*contention technique*) untuk mengirim pada kanal bersama. Protokol ALOHA dikembangkan untuk sistem satelit generasi awal adalah contoh terbaik dari teknik pengisian (*contention technique*). ALOHA mengatur pengguna memancar hanya jika ada data yang akan dikirim. Kemudian pengirim menunggu *feed back acknowledge*

(ACK) untuk menguji apakah pengirimannya berhasil atau gagal. Jika terjadi tabrakan data atau *collision*, pengirim menunggu beberapa saat untuk kemudian mengulang pengiriman paket data. Keunggulan dari teknik pengisian paket ini adalah kemampuannya melayani banyak pengguna tanpa khawatir terjadi *overhead*. Perkembangan selanjutnya protokol ALOHA tersebut digunakan sebagai dasar protokol-protokol standar IEEE 802.11 yang dikenal sebagai protokol *Ethernet* [1,2].

Pada suatu daerah yang tidak tersedia infrastruktur jaringan tetap, memerlukan banyak waktu dan biaya untuk menyediakannya, jaringan radio *ad-hoc* dapat dengan cepat melayani kebutuhan penyaluran data yang diperlukan. Dua node dalam jaringan *wireless ad-hoc* dapat saling berkomunikasi jika jarak antara keduanya kurang dari maksimum jangkauan *broadcast* dari masing-masing node radio. Semua node dalam jaringan radio *ad-hoc* harus berada dalam jangkauan transmisi satu dan lainnya, sehingga pergerakan node masih mungkin dipelihara ketersambungannya dalam jaringan menggunakan mekanisme *routing* konvensional *multi-hop*. Seperti contoh penggunaan jaringan tanpa infrastruktur yang biasanya dilakukan pada medan perang, daerah operasi SAR, konferensi dan sebagainya karena memiliki kemampuan untuk menangani kesalahan node radio dan perubahan topologi cepat. Gambar 1.1 menunjukkan contoh sederhana penggunaan jaringan radio *ad-hoc*, terdiri dari tiga komputer jinjing dengan kelengkapan radio. Jarak transmisi radio tidak memungkinkan untuk ketiganya saling berkomunikasi. Node A tanpa bantuan node B tidak mungkin berkomunikasi dengan node C. Node B dapat berkomunikasi dengan node A maupun node C, pada saat komunikasi terjadi antara node A dan node C saat itu node B berfungsi sebagai router bagi komunikasi tersebut.



Gambar 1.1 Jaringan radio *ad-hoc* sederhana

Tantangan yang ada adalah jika kemudian ada banyak node-node radio yang bergerak / *mobil* dalam jaringan radio *ad-hoc* untuk dapat tetap menjamin terjadinya komunikasi sampai pada node radio terjauh secara geografis. Kita perlu mencari tatacara rute yang terbaik agar komunikasi antara node-node radio tetap terjadi pada jaringan radio *ad-hoc* tersebut. Ada beberapa protokol rute pada jaringan radio *ad-hoc* yang *proactive* telah dikembangkan dan diakui oleh *Internet Engineering Task Force* (IETF) antara lain *Fisheye State Routing* (FSR) [4] dan *Optimized Link State Routing* (OLSR) [5].

1.2. Perumusan dan Batasan Masalah

Dalam sistem jaringan radio bergerak, dimana node-node radio *ad-hoc* yang *proactive* digunakan pada penelitian ini adalah computer jinjing/*laptop* dengan menambah peralatan radio berupa kartu *PC wireless* yang bekerja dengan protokol 802.11 sejumlah seratus node dalam area geografis 1000 x 1000 meter persegi, Simulasi yang dilakukan pada penelitian ini menggunakan Network Simulator 2.28, yang beroperasi pada *Operating System* Linux dan *Operating System* windows yang menggunakan program Cygwin akan diteliti dengan pola gerakan yang dibangkitkan untuk 5 waktu jeda yang berbeda: 0, 40, 100, 200 dan 400 detik. Dengan

menggunakan *ns-2* simulator sehingga dapat dievaluasi perbandingan unjuk kerja dari beberapa protokol rute jaringan radio *ad-hoc proactive* meliputi :

- a. Kemampuan untuk mencari routing sehingga didapatkan *throughput* terbaik dan waktu tunda tersingkat.
- b. Keakurasian data yang berhasil diteruskan dari node sumber ke node tujuan.
- c. End-to-End Delay

Hasil simulasi menggunakan *ns-2* simulator [3] digunakan untuk menentukan routing protokol untuk jaringan radio *ad hoc proactive* yang terbaik dari dua protocol yang diteliti :

- a. *Fisheye State Routing (FSR)*
- b. *Optimized Link State Routing (OLSR)*



1.3. Tujuan Penelitian

Penelitian ini bertujuan untuk mengetahui sejauh mana routing protokol *proactive* pada jaringan radio *ad hoc*, sehingga dapat memenuhi kebutuhan terjadinya komunikasi antar node yang jauh dengan *throughput* yang besar, waktu tunda terpendek, akurasi data baik, dan terjadi perubahan unjuk kerja routing kecil jika beban komunikasi/trafik naik. Pada akhirnya akan ditemukan routing protokol jaringan radio *ad hoc proactive* terbaik dari kedua routing protokol jaringan radio *ad hoc* yang diteliti.

1.4. Manfaat Penelitian

Penelitian ini mensimulasikan beberapa protokol rute jaringan radio *ad hoc* menggunakan simulator jaringan *ns-2*. Hasil simulasi digunakan untuk membandingkan protokol rute jaringan radio *ad hoc* terbaik, dari beberapa protokol

rute jaringan radio *ad hoc* yang diteliti. Sehingga pada saat hendak digunakan pada keperluan riil yang sesuai dengan kondisi penelitian dapat langsung dipilih protokol rute jaringan *ad hoc* terbaik sesuai kebutuhan.

1.5. Sistematika Penulisan

Secara keseluruhan thesis ini terdiri dari lima bab dan lampiran-lampiran dengan uraian bab masing-masing sebagai berikut:

- Bab I : Pendahuluan membahas Latar belakang, Perumusan dan Batasan Masalah, Sasaran, Tujuan Penelitian, Manfaat Penelitian dan Sistematika Penelitian.
- Bab II : Tinjauan Pustaka, Jaringan *wireless Ad Hoc*, meliputi tinjauan tentang penelitian yang telah dilakukan serta Protokol-protokol rute.
- Bab III : Model Simulasi, meliputi pembahasan *ns-2* simulator, Langkah-langkah simulasi, dan Evaluasi perbandingan unjuk kerja *routing* protokol *proactive*
- Bab IV : Hasil Simulasi, menguraikan hasil dari Unjuk kerja *routing* protocol *proactive* serta Perbandingan unjuk kerja routing Protokol.
- Bab V : Penutup menguraikan Kesimpulan Penelitian dan Saran-saran

BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dibahas tentang pustaka-pustaka yang berkaitan dengan *routing protocol proactive* jaringan radio *ad hoc*. Bahan yang ditinjau meliputi teks buku, tulisan ilmiah (paper), laporan ilmiah dan bahan-bahan yang berhubungan dengan *routing protocol*.

2.1. Jaringan Wireless

Jaringan wireless dibentuk oleh rute dan host. Dalam sebuah jaringan wireless, rute akan bertanggung jawab untuk penyampaian paket dalam jaringan dan host merupakan sumber atau tempat data mengalir. Perbedaan dasar antar wired dan jaringan wireless menjadi jalan untuk komponen jaringan komunikasi. Suatu *wired* jaringan bersandar pada kabel fisik untuk memindahkan data. Dalam suatu jaringan wireless, communication antara komponen jaringan terdapat perbedaan baik *wired* maupun *wireless*. Karena komunikasi wireless tidak mempunyai batasan kabel fisik, melainkan suatu kebebasan tertentu untuk host dan/atau rute dalam jaringan wireless bergerak. Ini adalah satu keuntungan dalam jaringan wireless.

Komponen jaringan wireless dapat berkomunikasi dengan menggunakan channel wireless. Perbedaan Frekwensi radio (RF) yang cakupan spektrum digunakan jaringan wireless, sebagai contoh, 27.5-29.5 GHZ untuk Lokal Multipoint Distribusi Sistem (LMDS), 2.5-2.7 GHZ untuk Multipoint Multichannel Distribusi Sistem, dan 5.15-5.35 GHZ dan 2.4-2.58 GHZ untuk IEEE 802.11a dan 802.11b. Kekuatan sebuah sinyal suatu medium *wireless* akan berkurang ketika pengiriman sinyal lebih lanjut. Ketika sebuah sinyal dikirimkan ke luar pada jarak tertentu, akan

mengurangi penguatan pada titik resepsi [6]. Jarak pengiriman sinyal dapat menjangkau titik ini apakah dapat menghubungi cakupan radio pada sinyal ini. Untuk menyederhanakan model transmisi ini, orang-orang berasumsi bahwa sinyal wireless cukup kuat untuk menerima sinyal jika penerima berada dalam cakupan radio.

Beberapa medium akses control (MAC) protokol digunakan dalam jaringan wireless guna untuk mengatur penggunaan pada medium *wireless*. Dalam Contoh meliputi *BLUETOOTH*, lapisan MAC protokol dan IEEE 802.11 MAC LAYER protokol. Sebab cakupan radio pada umumnya terbatas dan komponen jaringan mungkin mempunyai beberapa mobilitas, topologi suatu jaringan wireless akan berbeda menurut waktu.

2.2. Wireless Local Area Network (WLAN)

Dengan semakin bertambahnya pemakaian komputer, semakin besar kebutuhan akan pentransferan data dari satu terminal ke terminal lain yang dipisahkan oleh satuan jarak dan semakin tinggi kebutuhan akan efisiensi penggunaan alat-alat kantor (seperti printer dan plotter) dan waktu perolehan *data base*, maka semakin tinggi pula kebutuhan akan suatu jaringan yang menghubungkan terminal-terminal yang ingin berkomunikasi dengan efisien. Jaringan tersebut dikenal dengan *Local Area Network (LAN)* yang bias *Wireless LAN (WLAN)* merupakan suatu sistem komunikasi data tanpa kabel yang merupakan perluasan atau solusi alternatif dari LAN konvensional (dengan kabel). WLAN dapat dipasang dalam suatu gedung atau antar gedung, dengan menggunakan gelombang elektromagnetik untuk mengirim dan menerima data *Wireless LAN* juga dapat digunakan untuk

menjangkau wilayah LAN yang sulit dicapai dengan kabel tembaga biasa, dan juga untuk menjangkau pengguna bergerak memakai kabel atau fiber optik sebagai media transmisinya. Sesuai perkembangan karakteristik masyarakat seperti yang telah disebutkan di atas maka LAN menawarkan suatu alternatif untuk komputer portabel yaitu *Wireless LAN (WLAN)*. WLAN menggunakan frekuensi radio (RF) atau *infrared (IR)* sebagai media transmisi.

Wireless LAN (WLAN) merupakan suatu sistem komunikasi data tanpa kabel yang merupakan perluasan atau solusi alternatif dari LAN konvensional (dengan kabel). WLAN dapat dipasang dalam suatu gedung atau antar gedung, dengan menggunakan gelombang elektromagnetik untuk mengirim dan menerima data. *Wireless LAN* juga dapat digunakan untuk menjangkau wilayah LAN yang sulit dicapai dengan kabel tembaga biasa, dan juga untuk menjangkau pengguna bergerak.

Dibandingkan dengan LAN, WLAN memiliki kelebihan, antara lain:

- Mobilitas tinggi

WLAN memungkinkan pengguna untuk mengakses informasi secara real time dimanapun berada (dalam jangkauan WLAN), tidak terpaku pada satu tempat saja. Mobilitas yang tinggi tentunya dapat meningkatkan layanan dan produktivas.

- Kemudahan dan kecepatan instalasi

Instalasi WLAN sangat mudah dan cepat karena bisa dilakukan tanpa harus menarik dan memasang kabel melalui dinding ataupun atap.

- Fleksibel

Teknologi WLAN memungkinkan untuk membangun jaringan dimana kabel tidak dapat digunakan/tidak memungkinkan untuk digunakan.

- Menurunkan biaya kepemilikan

Meskipun biaya investasi awal untuk perangkat keras WLAN lebih mahal dari pada LAN konvensional, tetapi biaya instalasi dan perawatan jaringan WLAN lebih murah, sehingga secara total dapat menurunkan besar biaya kepemilikan.

- *Scalable*

WLAN dapat menggunakan berbagai topologi jaringan sesuai dengan kebutuhan, mulai dari jaringan independent yang hanya terdiri dari beberapa pengguna saja, sampai jaringan infrastruktur yang terdiri dari ribuan pengguna.

2.2.1 Arsitektur *Wireless Local Area Network*

Dalam membahas arsitektur suatu WLAN, terdapat dua penyusun dari arsitektur WLAN tersebut, yaitu Komponen WLAN dan Topologi WLAN. Berikut ini akan diuraikan kedua penyusun dari arsitektur WLAN.

2.2.1.1 Komponen-Komponen dalam WLAN

Komponen-komponen yang digunakan dalam menyusun suatu arsitektur WLAN, adalah sebagai berikut:

- *Access Point (AP)*, yaitu perangkat yang berfungsi sebagai penghubung dari satu pengguna ke pengguna lain atau dari suatu jaringan *wireless* ke jaringan *wireless* yang lain. Perangkat ini juga dapat berfungsi sebagai penghubung dari suatu lingkungan LAN ke *Internet Service Provider (ISP)*. Cara kerja dari perangkat ini adalah mengkonversikan sinyal radio (RF) menjadi sinyal digital yang akan disalurkan melalui kabel atau ke perangkat WLAN yang lain

dengan dikonversikan lagi menjadi sinyal radio. Contoh dari AP ditunjukkan oleh Gambar 2.1.

- *Wireless LAN Interface*, seperti ditunjukkan gambar 2.1. merupakan peralatan yang dipasang di *Access-Point* atau di *Mobile/Desktop PC*, peralatan yang dikembangkan secara massal adalah dalam bentuk PCMCIA *card*.



Gambar 2.1 Access Point dan Wireless LAN Interface [6].

- *Mobile/Desktop PC*, merupakan perangkat akses untuk pengguna, *mobile PC* pada umumnya sudah terpasang *port* PCMCIA sedangkan *desktop PC* harus ditambahkan *PC Card* PCMCIA dalam bentuk ISA (Industry Standard Architecture) atau PCI (Peripheral Component Interconnect) card.
- *Wired LAN*, merupakan jaringan kabel yang sudah ada, bila *Wired LAN* tidak ada maka hanya sesama *Wi-Fi* yang saling interkoneksi.
- *Antenna*, adalah perangkat yang berfungsi mentransmisikan sinyal radio dari *access point* ke *access point* yang lain atau ke perangkat *Wi-Fi* yang lain. Atau dari satu perangkat *Wi-Fi* ke *AP* atau ke perangkat *Wi-Fi* yang lain. Antena internal bisa mencapai jarak $\pm 300\text{m}$, sedangkan antena eksternal bisa mencapai puluhan Km.

Secara relatif perangkat Access Point (AP) ini mampu menampung beberapa sampai ratusan pengguna secara bersamaan. Beberapa vendor hanya merekomendasikan belasan sampai sekitar 40-an pengguna untuk satu AP. Meskipun secara teorinya perangkat ini bisa menampung banyak namun akan terjadi kinerja yang menurun karena faktor sinyal RF (*Radio Frequency*) itu sendiri dan kekuatan sistem operasi AP. Saat ini sistem operasi AP dikembangkan dengan dasar prosesor i486 dan RAM 4-8MB.

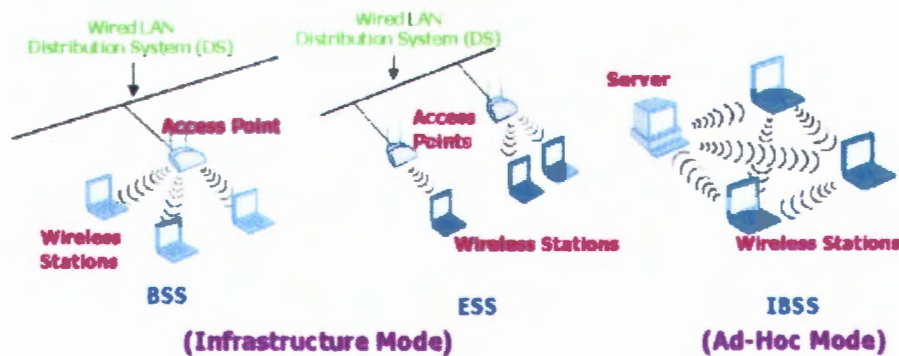
Komponen logika dari AP adalah ESSID (*Extended Service Set Identification*) yang merupakan standar dari IEEE 802.11. Pengguna harus mengkoneksikan PCMCIA *card*-nya ke AP dengan ESSID tertentu supaya transfer data bisa terjadi. ESSID menjadi autentifikasi standar dalam komunikasi *wireless*. Dalam segi keamanan beberapa vendor tertentu membuat kunci autentifikasi tertentu untuk proses autentifikasi dari klien ke AP.

Rawannya segi keamanan ini membuat IEEE mengeluarkan standarisasi *Wireless Encryption Protocol* (WEP), sebuah aplikasi yang sudah ada dalam setiap PCMCIA *card*. WEP ini berfungsi meng-*menencrypt* data sebelum ditransfer ke sinyal RF, dan meng-*decrypt* kembali data dari sinyal RF. Enkripsi yang umum dipakai adalah sebesar 40 bit dan ada beberapa vendor tertentu yang mengeluarkan WEP sampai 128 bit.

Selain komponen-komponen pokok di atas, beberapa *vendor* juga menawarkan komponen lain yang disesuaikan dengan keperluan yang diinginkan. Salah satu contohnya adalah *Wireless Router*, yang berfungsi seperti halnya AP, tetapi dapat digunakan untuk menghubungkan *subnet-subnet* yang berbeda.

2.2.1.2 Topologi Jaringan WLAN

WLAN memiliki beberapa topologi jaringan , yaitu *Basic Service Set (BSS)*, *Independent Basic Service Set*, dan *Extended Service Set (ESS)*.



Gambar 2.2 Topologi Jaringan WLAN [6].

Sebuah BSS terdiri dari beberapa stasiun yang berada dibawah kontrol langsung dari satu fungsi koordinat (DCF atau PDF). *Basic Service Area (BSA)* adalah area yang diliputi oleh BSS, atau bisa dianalogikan dengan sebuah sel dalam jaringan komunikasi seluler. Seluruh stasiun dalam sebuah BSS dapat berkomunikasi secara langsung dengan stasiun yang lain dalam sebuah BSS, tetapi harus melalui jaringan infrastruktur. Jaringan infrastruktur dibangun untuk penyediaan bagi pelanggan wireless dengan layanan khusus dan jarak yang jauh. Dalam IEEE 802.11 jaringan infrastruktur yang dibangun menggunakan sejumlah *Access Point (AP)*. Komunikasi dalam BSS bisa mengalami gangguan karena *multipath fading*, atau interferensi dari atau interferensi dari BSS yang berdekatan yang menggunakan karakteristik *physical layer* yang sama (seperti frekuensi dan kode *spreading* atau pola *hopping*), hal ini bisa menyebabkan beberapa stasiun nampak “tersembunyi” dari stasiun-stasiun yang lain.

Sebuah jaringan IBSS terdiri dari sekumpulan stasiun-stasiun dalam sebuah BSS untuk tujuan komunikasi *internetwork* tanpa bantuan dari infrastruktur jaringan. Beberapa stasiun dapat membangun komunikasi langsung dengan stasiun yang lain dalam BSS, tanpa membutuhkan penyaluran trafik melalui sebuah *Access Point*(AP). Topologi ini dikenal juga sebagai topologi *ad hoc*.

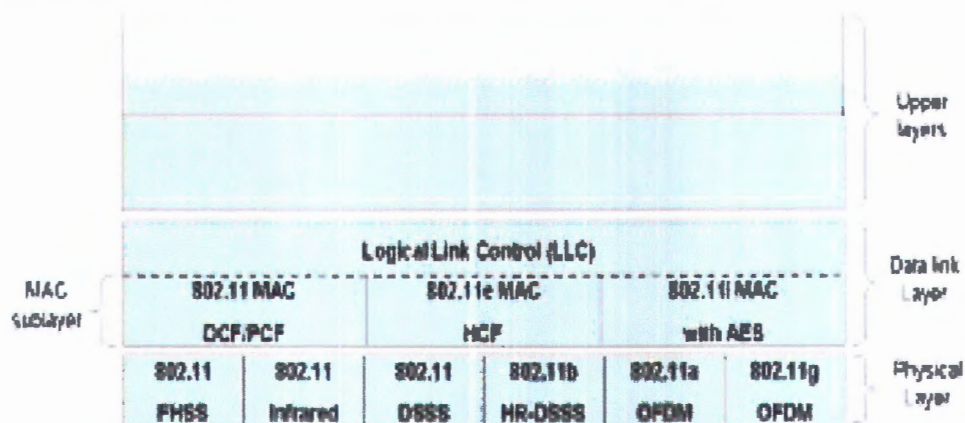
Extended Service Set (ESS) terdiri dari dua atau lebih BSS yang membentuk satu subnetwork yang dihubungkan dengan sebuah *Distribution System* (DS). DS dapat dianalogikan sebuah backbone jaringan yang digunakan untuk transport dari *MAC Service Data Units* (MSDUs) ke level MAC. DS dapat berupa sebuah kabel IEEE 802.3 *Ethernet* LAN, IEEE 802.4 *token bus* LAN, IEEE 802.5 *token ring* LAN, *Fiber Distributed Data Interface* (FDDI) *Metropolitan Area Network* (MAN), atau media *wireless* IEEE 802.11 yang lain. DS digunakan sebagai transport backbone untuk mentransfer paket-paket antara BSS yang berbeda-beda dalam sebuah ESS. Pada ESS, masing-masing BSS dihubungkan dengan AP (dalam hal ini dianalogikan sebagai *Base Station*). Seluruh AP dihubungkan ke *Distribution System* (DS) yang hampir mirip dengan sebuah backbone, biasanya Ethernet atau wireless.

Sebuah ESS dapat juga menyediakan akses gateway untuk pengguna *wireless* ke jaringan *wired* misalnya internet. Alat yang menghubungkan pengguna *wireless* dan *wired* dikenal dengan sebutan portal. Portal ini berhubungan dengan DS, yang akan menghubungkan jaringan IEEE 802.11 dan jaringan non-IEEE 802.11.

2.2.2 Layer-Layer dalam WLAN

WLAN memiliki dua *layer* pokok dalam melakukan komunikasi data. Dua *layer* tersebut adalah *Medium Access Control* (MAC) *sublayer* dan *physical* (PHY)

layer. IEEE 802.11 menempatkan kedua layer tersebut ke dalam *task group* yang berbeda menurut *Open System Interconnection*, sebagai berikut:



Gambar 2.3 Posisi Layer WLAN dalam Model OSI [6].

PHY layer terletak pada layer 1 (*Physical layer*), sedangkan MAC menduduki sebagian dari *Data link layer*, oleh sebab itu disebut *sublayer*.

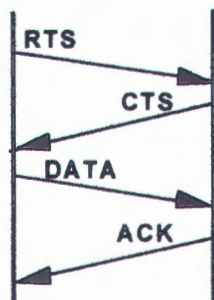
2.3. Media Pengaksesan

2.3.1. Medium Access Control Protocol

Jaringan komputer menggunakan kabel kebanyakan menggunakan protokol *Media Access Controller* (MAC) yakni *Carrier Sense Multiple Access Collision Detection* (CSMA/CD). Alokasi bandwidth user diatur dan dilayani menggunakan MAC protokol [7]. Protokol ini digunakan untuk mengatur penggunaan kanal bersama dengan cara mendengarkan aktifitas media sebelum mengirimkan data, carrier sense. Jika sebuah titik jaringan (stasiun N) sudah mengirimkan data, maka stasiun (A) yang hendak mengirim menunggu sampai node N selesai menggunakan kanal. Jika secara bersamaan dua stasiun mengirimkan data, maka akan terjadi tabrakan data dan informasi yang dikirim keduanya rusak. Deteksi tabrakan data, *collision detection*, akan mengatasi kecelakaan ini dengan cara mendengarkan sinyal

sebelum stasiun mulai mengirimkan data. Saat terjadi tabrakan data, stasiun pengirim menghentikan proses pengiriman untuk kemudian mencoba mengirim ulang, setelah mengoperasikan algoritma *backoff*. Teknologi yang tata cara pengaksesan media kabel (CSMA/CD) sulit untuk dilakukan pada media udara atau jaringan tanpa kabel.

Protokol pengaksesan media *wireless* hampir sama dengan media kabel yaitu *Carrier Sense Multiple Access Collision Avoidance* (CSMA/CA atau MACA) [56, 61]. Pada jaringan *wireless* kondisi normal, semua stasiun melakukan mekanisme standar untuk menghindari terjadinya tabrakan data informasi *wireless*. Sinyal radio tidak bisa dideteksi jika tabrakan data / *collision* terjadi, karenanya stasiun-stasiun mencoba untuk menghindari *collision* dengan menunggu media udara / *wireless* menjadi bersih untuk beberapa saat hingga dapat mengirimkan paket data melalui media udara (misalnya, untuk paket data yang harus dikirim ke stasiun terjauh)

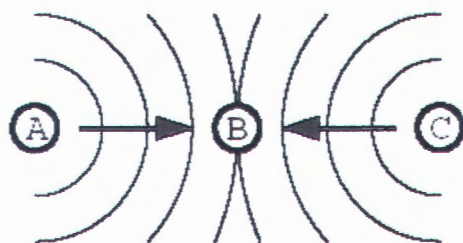


Gambar 2.4. Sinkronisasi 4 langkah CSMA/CA

Bila kondisi media udara bersih suatu stasiun yang akan mengirimkan paket-paket data mengirimkan sinyal paket kecil *request to send* (RTS) berisi alamat tujuan dan durasi pancaran yang direncanakan ke stasiun tujuan. Bila stasiun tujuan menerima sinyal RTS dan dari pengamatannya kondisi media bersih, ia akan mengirimkan jawaban sinyal paket kecil *clear to send* (CTS). Setelah menerima sinyal CTS, maka stasiun pengirim akan mengirimkan informasi data sesungguhnya

ke stasiun tujuan. Paket demi paket data dikirimkan sampai selesai sambil menguji melalui jawaban *acknowledge* (ACK) tiap paket dari stasiun penerima. Bila stasiun pengirim tidak menerima sinyal CTS dari stasiun tujuan, ia akan melakukan pengulangan prosedur dengan mengirim lagi sinyal RTS ke stasiun tujuan. Proses mekanisme menunggu media udara bersih dengan cara mendengarkan sinyal-sinyal akan dimulai lagi setelah batas waktu pengiriman terlampaui.

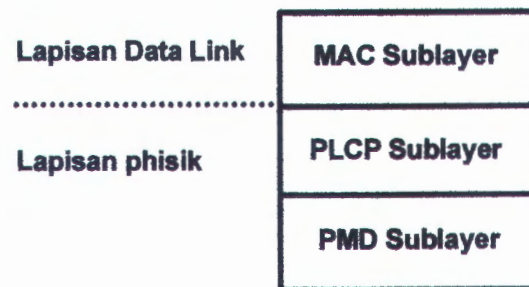
Pada banyak kasus protokol ini cukup berguna mengurangi tabrakan data, karenanya meningkatkan utilitas bandwidth. Akan tetapi sering masih belum mengatasi pada problem stasiun tersembunyi (*hidden station*). Yakni stasiun yang letaknya cukup jauh dari stasiun-stasiun lain yang sulit untuk “mendengar” stasiun lain dengan cepat, karena waktu yang digunakan untuk mengirimkan paket dari stasiun pengirim ke stasiun tujuan lebih besar dari waktu tunggu stasiun tujuan. Gambar 2.5 menunjukkan problem *hidden station*, stasiun C tidak bisa mendengar stasiun A. Jika stasiun A mengirim sinyal ke stasiun C, stasiun C tidak dapat menerima. Atau jika dapat menerima sinyal RTS dari stasiun A, saat mengirim CTS stasiun A tidak mendengar. Hal ini akan mengakibatkan stasiun B menganggap ada *collision* di media.



Gambar 2.5. Problem stasiun tersembunyi/hidden station [24].

2.3.2. Lapisan Fisik 802.11

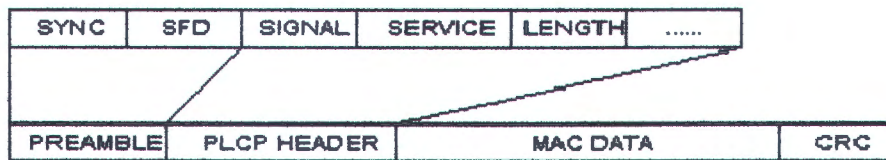
Lapisan fisik pada protocol 802.11 terbagi menjadi 2 sub lapis yakni : sub lapisan *Physical Layer Convergence Procedure* (PLCP) dan sub lapis *Physical Media Dependent* (PMD). Sub lapis PLCP mengadaptasi kemampuan *physical medium dependent system* menjadi layanan lapisan fisik. PLCP menetapkan suatu metoda pemetaan 802.11 PHY sub lapis *Service Data Unit* (PSDU) ke dalam format frame yang sesuai untuk mengirim atau menerima data dan pengaturan informasi antara dua atau lebih stasiun yang bersama-sama menggunakan *physical medium dependent system*. Hal ini menjadikan 802.11 MAC dapat beroperasi dengan ketergantungan minimum pada sub lapis PMD.



Gambar 2.6. Model referensi IEEE 802.11 [8].

Sub lapis PMD menetapkan karakteristik dan metoda pengiriman dan penerimaan data melalui media *wireless* antara dua atau lebih stasiun-stasiun yang menggunakan sistem modulasi yang sama [8].

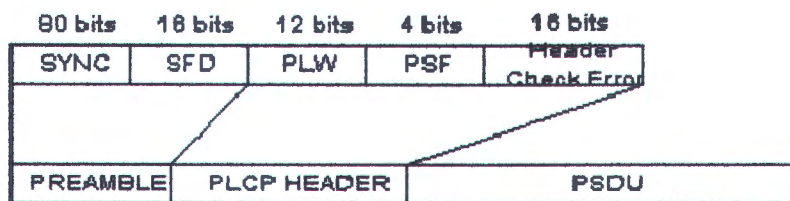
Frame PLCP dibuat menggunakan komponen-komponen secara umum seperti ditunjukkan pada gambar 2.7 berikut ini:



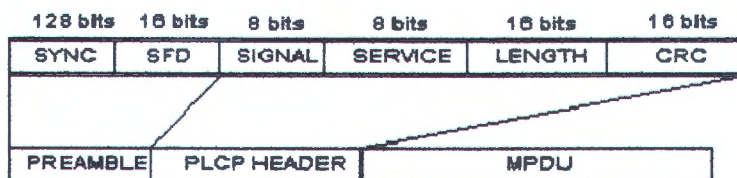
Gambar 2.7. Format frame PLCP secara umum

Preamble tergantung pada PHY. Ia terdiri dari dua bagian : sinkronisasi (SYNC) dan *Start Format Delimiter* (SFD). SYNC adalah sebuah deretan campuran zeros dan ones, yang digunakan oleh rangkaian PHY untuk memilih antenna yang tepat, dan untuk memenuhi kondisi *steady-state* demodulasi serta penyinkron bit-bit clock. SFD digunakan untuk menentukan *frame timing*.

PLCP ditransmisikan pada 1 Mbps dan berisi informasi logic yang digunakan oleh lapisan PHY untuk menkode frame. *Field* pensinyalan PLCP berisi informasi laju bit. *Field Service* adalah field cadangan untuk penggunaan memenuhi perkembangan yang akan datang. *Filed Length* menunjukkan panjang data MAC atau jumlah waktu (mikrosekon) yang diperlukan untuk mengirim data MAC.



Gambar 2.8. Frame PLCP untuk FHSS



Gambar 2.9. Frame PLCP untuk DSSS [24].

Frame PLCP untuk FHSS (gambar 2.9) menunjukkan field SYNC *preamble* terdiri dari 80 bit berisi alternatif pola *zero-one*, pengiriman diawali dengan *zero* dan diakhiri dengan *one*. Field SFD berisi pola biner 16 bit 0000 1100 1011 1101 (*transmitted leftmost bit first*),. Bit pertama dari SFD mengikuti bit terakhir pola SYNC. Field PLW (*PSDU length word*) menunjukkan jumlah oktet yang terdapat dalam PSDU. 4-bit PSF (*PLCP signaling field*) mendefinisikan ukuran laju bit antara 1 Mbps sampai 4,5 Mbps dengan tahap kenaikan 0,5 Mbps. Bit pertama dari PSF disiapkan untuk perkembangan pemakaian yang akan datang. Frame PLCP untuk DSSS (gambar 2.6), field SYNC bisa terdiri dari 128 bit. SFD terdiri dari 16 bit '0XF3A0' 1111 0011 0101 0000. Field SIGNAL menunjukkan laju data [29] :

a.	'0X0A'	0000 0101	1 Mbps
b.	'0X14'	0001 0100	2 Mbps
c.	'0X37'	0011 0111	5,5 Mbps
d.	'0X6E'	0110 1110	11 Mbps

Preamble dan *header* PLCP dikirimkan pada laju data 1 Mbps, sedangkan data informasi dikirimkan dengan laju data salah satu dari ke-empat kemungkinan kecepatan. 1 Mbps, 2 Mbps, 5,5 Mbps atau 11 Mbps. Pada laju transmisi data 1 dan 2 Mbps sinyal DSSS dibuat menggunakan sekuensi *spreading* tetap dibentuk melalui 11 chip pengkode *Baker*. Untuk laju 1 Mbps sekuensi *spreading* tetap digunakan untuk menyebarkan 1 Mbps sinyal *binary phase shift keying* (BPSK), untuk 2 Mbps, dengan cara penyebaran yang sama adalah modulasi *quaternary PSK* (QPSK). Untuk semua laju data 1 – 11 Mbps, laju chip adalah 11 Mchips/s [9].

2.4. Protokol Rute (*Routing Protocol*)

Sebuah protokol rute diperlukan jika pada proses komunikasi data paket dikirimkan melalui satu atau beberapa titik / node untuk mencapai stasiun tujuan. Protokol rute bertugas untuk menentukan rute perjalanan paket data menuju ke alamat tujuan. Saat ini terdapat dua jenis protokol rute yakni jenis *distance vector* dan jenis *link state*. Kedua jenis protokol rute tersebut umumnya digunakan pada jaringan dengan infrastruktur statis, perubahan topologi yang sangat dinamis tidak menjadi acuan saat kedua jenis protokol tersebut dibuat. Secara nyata keduanya tidak dapat digunakan pada protokol routing *non adaptif* untuk keperluan jaringan yang node-node-nya bergerak bebas. Keduanya secara periodik mengirimkan pesan kontrol tentang kondisi topologi jaringan, kerjanya akan menjadi buruk jika digunakan pada jaringan yang besar dan proses pengiriman data melalui rute yang panjang. Untuk memenuhi keperluan tersebut diperlukan banyak pesan kontrol rute dan pada topologi bergerak / mobil perodesasi pengiriman pesan kontrol rute akan semakin meningkat. Hal ini akan mengakibatkan terlalu banyak lalu-lintas / trafik pesan kontrol rute yang menyita *bandwidth*.

2.4.1. Studi Literatur

Literatur yang membahas masalah teknik *routing protocol* terdapat banyak sekali. Literature [10] membahas masalah keuntungan routing lintasan terpendek yang memudahkan manajer jaringan menangani beban pemeliharaan. Sekali lintas terpendek didapat semua rute secara implicit terdefinisi. Jika suatu lintasan rute yang panjang terjadi kerusakan, jaringan secara otomatis akan mengatasi kesalahan, memeriksa apakah masih bisa dilakukan. Algoritma untuk mendapat lintasan terpendek berdasarkan teori graph dibahas pada literature [11, 12] dengan beberapa

cara seperti Dijkstra's dan Bellman-Ford. Sedangkan jaringan dan teknik multiple access banyak dibahas pada literature [13, 14, 15]

2.4.1.1. Penelitian Terdahulu

Selain dari studi literature, menganalisa tentang routing protocol proactive pada jaringan wireless ad hoc, yang telah banyak dilakukan oleh para peneliti antara lain :

1. **Xiaoyan Hong, Kaixin Xu, and Mario Gerla, [3]** mepresentasikan penemuan tentang routing protocol jaringan ad hoc. Protocol ini bekerja dalam tiga katagori yaitu flat routing protokol, pendekatan hierarchical, dan GPS Geographical routing. Makalah mereka membandingkan berbagai routing protocol tersebut beserta bentuk operasionalnya yang digunakan routing protocol dalam jaringan radio ad hoc. Dalam protocol ini, rute akan menghitung exsploitasi adalah secara terbatas. FSR (Fisheye State Routing) akan mengurangi banyaknya traffik dalam route untuk menyesuaikan frekuensi dengan rute yang baru. OLSR mengurangi ukuran paket rute dan banyaknya penyampaian paket dalam node. TBRPF akan membatasi pengembangan tentang rute yang baru pada cabang node dan melaporkan diferensial informasi pada sumber. Kedua-Duanya OLSR dan TBRPF bekerja lebih efisien dalam jaringan yang padat, sedangkan FSR lebih pantas digunakan untuk diameter jaringan yang besar.
2. **Wenbo Fu and Dr. Stephan,** dalam makalahnya [16] mepresentasikan penemuan protocol baru tentang MANET (Mobile Ad hoc Network) yaitu protokol proactive dan reactive. Dalam penelitian ini akan menjelaskan

tentang perbandingan unjuk kerja dari masing-masing protocol proactive dan reactive yaitu protocol OLSR, TBRPF, AODV dan DSR. Procedure yang digunakan untuk routing protocol proactive dan protocol reactive adalah untuk menyesuaikan pada karakteristik MANET (Mobile Ad hoc Network). Algoritma routing Link State digunakan untuk menandakan link atau tetangga yang jauh dari node, algoritma Dijkstra'S, digunakan untuk mengkalkulasi alur yang paling pendek kepada node yang tersedia dalam jaringan.

3. **Wismanu Susetiyo**, dalam makalahnya [17] telah melakukan penelitian tentang Evaluasi Unjuk kerja Protokol Rute pada jaringan wireless Ad hoc Multi-hop, mempresentasikan tentang Ad hoc routing protokol Reactive (On-demand), seperti DSR (Dynamic Source Routing), AODV (Ad hoc On-Demand Distance Vector), DSDV (Destination – Sequenced Distance Vector) dan TORA (Temporally Ordered Routing Algorith). Penelitian ini dilakukan menggunakan level paket pada 50 node bergerak untuk mengevaluasi dan membandingkan unjuk kerja ke empat protokol rute pada jaringan wireless ad hoc multi-hop. Hasil simulasi yang dilakukan dapat diamati, dimana masing-masing protocol akan terjadi perubahan gerakan node dalam jaringan dan saat terjadi perubahan beban kerja jaringan.
4. **J.J. Garcia Lunes Aceves**, dalam makalahnya [11] mempresentasikan, menganalisa dan memvalidasi keluarga algoritma lintas terpendek untuk kumpulasi dinamis dalam jaringan kumputer atau internet. Algoritma yang mempresentasikan ini untuk mengatasi problem routing lintas terpendek terdistribusi menggunakan komputasi dufusi yang pertama sekali dipresentasikan oleh Dijkstras dan Scholten. Algoritma ini mempersingkat



waktu konvergensi setelah terjadinya satu perubahan cost link atau topologi jaringan untuk menjadikan bebas dari kondisi berputar-putar (loop-free). Komputasi difusi ini dimulai saat sebuah node tumbuh dengan mengirim pertanyaan-pertanyaan dan menyusut setelah menerima jawaban sepanjang suatu acyclic graph rooted pada sumber dari komputasi. Algoritma itu sendiri dapat digunakan untuk mengkonstruksi graph acyclic.

Faktor negatif lain yang terjadi pada stasiun bergerak, mereka menggunakan *CPU* untuk mengirim dan menerima pancaran radio, bahkan pada saat *stand by*, yang akan menguras energi dari baterai atau sumber tegangan. Semua protokol rute konvensional yang digunakan pada jaringan dengan media kabel berasumsi bahwa rute-rute dilakukan pada dua arah dengan kualitas yang sama. Sedangkan pada jaringan tanpa infrastruktur atau jaringan *ad hoc* asumsi tersebut tidak selalu benar. Prototol-protokol rute dapat diklasifikasikan menjadi tiga kategori:

- Terpusat atau tersebar
- Adaptif atau statis
- Reaktif atau proaktif atau *hybrid*

Jika kerja protokol rute dilakukan secara terpusat, semua keputusan yang dibuat ditentukan dari titik terpusat, tidak demikian dengan protokol rute terdistribusi, semua node atau titik melakukan kerja sama untuk memutuskan jalur rute pengiriman data. Suatu protokol rute adaptif mungkin mengubah perilaku kerjanya menyesuaikan dengan status jaringan, mengantisipasi kemungkinan terjadi kongesti pada sebuah link atau karena banyak faktor yang mungkin terjadi. Sedangkan suatu protokol rute reaktif melakukan aksi-aksi yang diperlukan misalnya

pencarian rute-rute bila diperlukan, kebalikannya protokol rute proaktif menemukan rute-rute sebelum mereka kemudian digunakan. Metoda reaktif juga disebut sebagai *on-demand routing protocol*. Karena aksinya dilakukan saat diperlukan, beban pengontrolan paket secara signifikan ditekan. Metoda proaktif menyimpan tabel rute, dan merawat tabel-tabel tersebut secara periodik. Metoda *hybrid* menggunakan kedua metoda reaktif dan proaktif tadi untuk membuat protokol rute semakin efisien. Contoh penggunaan metoda *hybrid* adalah protokol rute Zona.

2.5. Rute Protokol Proaktif

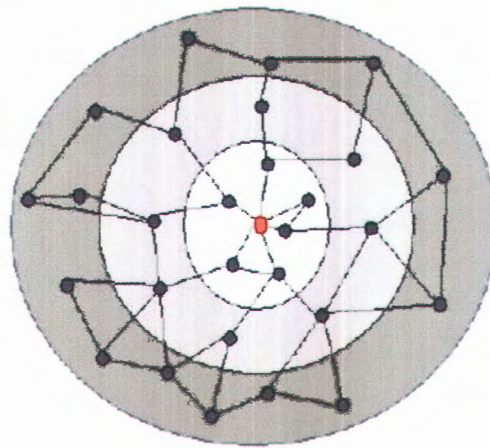
Ruting Protokol proaktif di bagi suatu bentuk umum, adalah sumber informasi rute akan menukar dengan menurut permintaan komunikasi. Protokol ini layak mempunyai terutama untuk aplikasi yang mencakup real-time pada jaminan komunikasi dan Qos, seperti *low-latency* rute yang mengakses dan mengubah jalur Qos mendukung dan monitoring. Banyak rute protokol proaktif telah diusulkan untuk efisiensi jaringan wireless, terutama pada jaringan wireless ad hoc.

2.5.1. OLSR (Optimized Link State Routing)

Optimized Link State Routing Protocol (OLSR) [15] adalah suatu protocol Link State (LS). Pada waktu tertentu topologi terjadi pertukaran informasi dengan node dalam jaringan. Protokol ini menggunakan *multipoint relays* (MPR) [16] guna untuk Mengurangi banyaknya pengiriman paket dan juga ukuran paket terbaru dari LS, pengendalian banyaknya pesan dalam suatu jaringan. Sebagai hasilnya, metoda untuk pengendalian pengiriman pesan dalam jaringan harus mencukupi kondisi-kondisi tertentu, pertama, pesan harus mampu menjangkau semua node lain. kedua, tidak ada sistim pengiriman kembali dalam routing protocol OLSR.

2.5.1.1. Fisheye State Routing (FSR)

Rute mata ikan / *fisheye* [57] adalah pengembangan dari GSR. Jumlah pesan pembaruan pada GSR menghabiskan cukup banyak bandwidth jaringan. Pada FSR pesan pembaruan tidak berisi beberapa informasi tentang keseluruhan node. Informasi lebih banyak berisi tentang node-node terdekat daripada node-node yang jauh, akan memperkecil ukuran pesan pembaruan. Tiap node mendapat informasi akurat tentang tetangganya, keakurasian informasi semakin berkurang, untuk node-node yang jaraknya makin jauh.



Gambar 2.10 Akurasi informasi pada FSR [24]

Gambar 2.10. menunjukkan lingkup lingkaran dari mata ikan untuk sebuah node pusat. Sebuah lingkup jangkauan didefinisikan secara terminologi sebagai sebuah node yang dapat dicapai melalui hop-hop tertentu. Node pusat memiliki informasi paling akurat tentang semua node pada lingkup lingkaran putih. Ia tidak mengetahui dengan benar tentang node-node, tetapi seperti sebuah paket bergerak lebih dekat ke tujuan, akurasi meningkat, dan paket disampaikan dengan benar. FSR dapat cocok digunakan pada jaringan *ad hoc* agak besar.

2.5.2. Prinsip Dasar Protokol Rute

Algoritma rute dapat dibedakan berdasarkan pada beberapa karakteristik kunci. Pertama, tujuan utama dari perancang algoritma mempengaruhi operasi hasil kerja protokol rute. Kedua, sudah ada beberapa tipe algoritma rute, dan masing-masing algoritma memiliki pengaruh yang berbeda pada jaringan dan sumber daya router. Akhirnya, algoritma rute menggunakan berbagai metrik yang mempengaruhi perhitungan optimasi perutean. Bagian berikut menguraikan beberapa protokol rute yang merupakan dasar bagi pengembangan algoritma protokol-protokol rute baru.

2.5.2.1. Distance Vector Routing

Protokol rute ini menjaga tabel rute berisi lintasan untuk pengiriman keluarannya, hanya menginformasikan *link cost* yang diestimasi ke router tetangga saja dengan cara menyebarkan tidak dengan cara mem"*flooding*" [8]. Semua node / titik router menghitung lintasan terpendek ke tujuan yang diinginkan menggunakan informasi sebaran yang diterima. Dengan dasar kerja tersebut protokol ini termasuk jenis protokol terdistribusi, dengan begitu diharapkan jika ada banyak node / titik router yang berkontribusi di jaringan akan didapatkan ketepatan dan perkiraan lintasan terpendek untuk menyalurkan paket data [16]

2.5.2.2. Link State Routing

Protokol rute ini menjaga tabel rute yang berisi topologi lengkap yang telah dibangun untuk mendapatkan lintasan terpendek melalui *link cost*. Secara periodik informasi *link cost* diterima dan dikirimkan oleh semua node / titik router menggunakan teknik *flooding* [8]. Tiap node / titik router memperbarui table rutanya berdasarkan informasi yang telah diterima. Informasi *link cost* sangat mungkin

tidak konsisten karena sifat topologi pada medium udara / radio, seperti waktu tunda karena propagasi. Cara ini mungkin mengakibatkan *routing loop* yang agak panjang sampai mereka semua dapat memulihkannya melalui pembaruan link.

2.5.2.3. Source Routing

Protokol rute ini meletakkan informasi rute pada setiap header paket data yang dikirimkan. Penentuan rute dibuat saat data diberangkatkan. *Routing loop* dapat dihindari, meskipun kerja protokol menjadi tidak praktis. Untuk topologi yang berubah dengan cepat hal tersebut menjadi tidak efisien, protokol menjadi tidak akurat karena kesalahan rute selama pengiriman paket.

2.5.3. Sifat-sifat Protokol rute jaringan radio ad-hoc yang diharapkan

Secara ideal protokol rute pada jaringan ad-hoc diharapkan memiliki sifat-sifat sebagai berikut:

- Terdistribusi, untuk keperluan menaikkan realibilitas. Bila semua node/stasiun bergerak / mobil, sangat tidak cocok jika digunakan protokol rute terpusat. Tiap node / stasiun harus memiliki kecerdasan menentukan rute yang harus ditempuh menggunakan informasi node-node yang berdekatan. Protokol rute terdistribusi tetapi secara virtual terpusat adalah sebuah pemikiran yang baik.
- Asumsi link ke segala arah. Media *wireless* secara fisik memungkinkan pembukaan link ke segala arah. Tidak hanya melakukan komunikasi ke dua arah. Karena itu protokol rute juga harus dirancang untuk dapat menentukan rute link ke segala arah.

- Efisien dalam penggunaan daya listrik. Protokol rute yang memperhitungkan efisiensi daya. Karena komputer jinjing menggunakan baterai yang kecil, misalnya komputer palm. Protokol sebaiknya memperhitungkan beban penyebaran pesan kontrol rute, karena bila sebuah node / stasiun mati topologi akan berubah. Hal ini akan mengakibatkan rute tidak terakses. Karena itu diharapkan menggunakan pola *multiple* rute.
- Keamanan cukup terjamin. Sifat keamanan pada jaringan ad-hoc kurang baik. Media *wireless* yang digunakan sangat terbuka untuk diserang para pengganggu. Pada lapisan fisik serangan perusakan layanan, DoS (*denial of service*), dapat dihindari menggunakan cara pengkodean dan hopping frekuensi spread spectrum. Akan tetapi pada *level routing*, lapisan network, diperlukan autentikasi untuk komunikasi antar node-node, penerimaan dan enkripsi diperlukan untuk menghindari pengguna router liar pada jaringan pribadi.
- Protokol rute *hybrid*. Sebuah protokol yang lebih reaktif tidak hanya proaktif untuk menghindari kerja berat protokol.
- Menjaga kualitas layanan, QoS. Protokol sebaiknya tahu waktu tunda dan throughput antara sumber ke tujuan dan dapat mengujinya sehingga aplikasi *real-time* bisa diteruskan dengan baik.

2.5.3.1. Quality of Service

Quality of Service (QoS) adalah suatu pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari

suatu servis/layanan. QoS biasanya digunakan untuk mengukur sekumpulan atribut performansi yang telah dispesifikasikan dan biasanya diasosiasikan dengan suatu servis. Pada jaringan berbasis IP, QoS mengacu pada performansi dari paket-paket IP yang lewat melalui satu atau lebih jaringan.

QoS didesain untuk membantu *end user* menjadi lebih produktif dengan memastikan bahwa dia mendapatkan performansi yang handal dari aplikasi-aplikasi berbasis jaringan. QoS mengacu pada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda-beda. QoS merupakan suatu tantangan yang cukup besar dalam jaringan berbasis IP dan Internet secara keseluruhan. Tujuan dari QoS adalah untuk memuaskan kebutuhan-kebutuhan layanan yang berbeda, yang menggunakan infrastruktur yang sama. QoS menawarkan kemampuan untuk mendefinisikan atribut-atribut layanan jaringan yang disediakan, baik secara kualitatif maupun kuantitatif. Adapun aspek-aspek yang perlu diperhatikan pada QoS ini adalah sebagai berikut :

2.5.3.2. Delay

Ada 2 macam tipe *delay* yang akan memiliki dampak berbeda-beda terhadap suatu layanan, seperti :

- *end-to-end delay (latency)*
- variasi *delay* atau *jitter*

Untuk layanan sendiri terdiri atas beberapa macam, akan tetapi pada umumnya dapat dikelompokkan menjadi 3 golongan besar, antara lain layanan aplikasi yang bersifat *real time* interaktif, *real time* non-interaktif, dan layanan aplikasi non *real time*.

- Aplikasi *real time* interaktif, seperti komunikasi suara atau yang terbaru – *video conference*, sangat sensitif terhadap *end-to-end delay* dan *jitter*. *Delay* yang panjang dapat mengurangi interaktifitas dalam berkomunikasi.
- Aplikasi *real time* non-interaktif, seperti *broadcast* satu arah, kurang dipengaruhi oleh *end-to-end delay* tetapi sangat dipengaruhi oleh *jitter*. *Jitter* biasanya ditimbulkan akibat dari penggunaan suatu *buffer* pada *receiver* karena paket yang diterima disimpan dan diputar kembali pada waktu *offset* yang bersesuaian.
- Aplikasi non *real time*, pada umumnya tidak sensitif terhadap *delay*. Walaupun begitu, karena aplikasi ini dapat digunakan untuk pengukuran *delay* dalam mengontrol laju trafik (misalnya *Transmission Control Protocol* - TCP) atau kemungkinan menampung data sampai mendapat *acknowledge* (misalnya *file transfer protocol* - FTP), maka besar atau variabel *delay* dapat mempengaruhi kualitas dan kecepatan aplikasi-aplikasi tersebut.

Berikut ini beberapa macam komponen *end-to-end delay* :

- *Delay* transmisi : waktu yang dibutuhkan untuk meletakkan seluruh bit dari suatu paket ke dalam *link*.
- *Delay* propagasi : waktu yang dibutuhkan suatu bit untuk melewati suatu *link*.
- *Delay* proses : waktu yang dibutuhkan untuk memproses suatu paket dalam suatu elemen jaringan (misalnya merutekan paket pada output *port*).
- *Delay* antrian : waktu dimana suatu paket harus menunggu dalam antrian sebelum suatu paket dijadwalkan untuk ditransmisikan.

Terdapat juga beberapa *delay* tambahan ketika melewati paket dari suatu *interface* jaringan ke aplikasi yang akhirnya menuju ke *user*, seperti *delay* pada pentransferan paket melewati *bus host*, *delay* pada saat pengandaan paket dari ruang kernel ke ruang pengguna, dan *delay* pada penjadwalan aplikasi.

2.5.3.3. Throughput

Throughput adalah jumlah bit yang ditransmisikan per detik melalui sebuah sistem atau media komunikasi. *Throughput* diukur setelah transmisi data, karena suatu sistem akan menambahkan *delay* yang disebabkan *processor limitations*, kongesti jaringan, *buffering inefficiencies*, *error* transmisi, *traffic loads* atau mungkin desain *hardware* yang tidak mencukupi. Aspek utama *throughput* yaitu berkisar pada ketersediaan *bandwidth* yang cukup untuk menjalankan suatu aplikasi. Hal ini menentukan besarnya trafik yang dapat diperoleh suatu aplikasi saat melewati jaringan. Aspek penting lainnya adalah *error* (pada umumnya berhubungan dengan *link error rate*) dan *losses* (pada umumnya berhubungan dengan kapasitas *buffer*).

Nilai *throughput* tergantung pada faktor-faktor berikut ini :

- karakteristik *link* : *bandwidth*, *error rate*.
- karakteristik *node* : kapasitas *buffer*, daya pemrosesan.

Tabel berikut ini menunjukkan tingkat kesensitifan berbagai jenis trafik terhadap variabel-variabel performansi.

BAB III

METODOLOGI PENELITIAN

3.1. Model Simulasi

Untuk membandingkan unjuk kerja *routing protocol proactive* pada jaringan *wireless ad hoc* multihop yang dievaluasi, digunakan simulator *ns-2* [21]. Bab III ini akan menjelaskan tentang simulator *ns-2* dan menentukan metode yang digunakan untuk simulasi.

3.1.1. Network Simulator

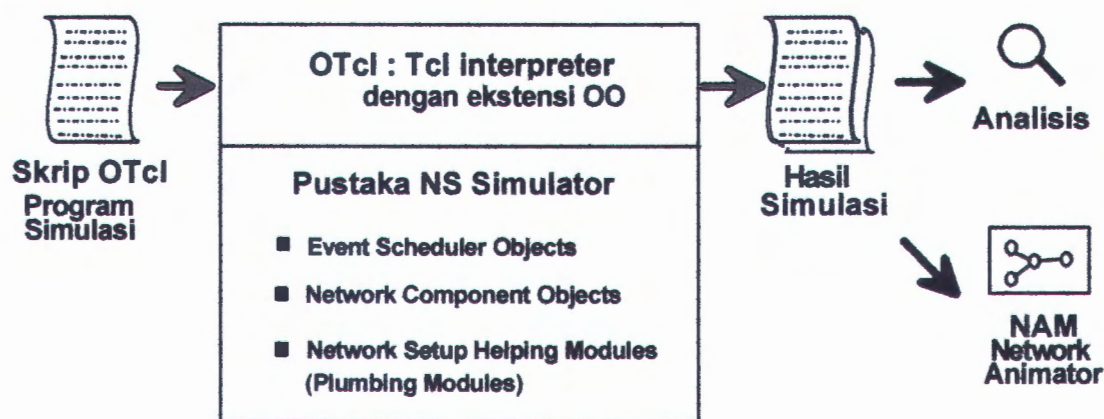
Network Simulator (NS) merupakan suatu media simulasi yang pada dasarnya bekerja pada system Unix / Linux. Agar dapat menjalankan NS bisa menggunakan OS Linux ataupun Windows. Akan tetapi agar dapat dijalankan pada system Windows harus menggunakan Cygwin sebagai linux environment-nya. Kelebihan NS adalah, pada NS terdapat tool validasi. Tool validasi ini digunakan untuk menguji validitas pemodelan yang ada pada NS. Secara default semua pemodelan NS akan melewati proses validasi ini. Jika kita ingin melakukan validasi pada pemodelan protocol yang ada pada library NS, kita dapat mengetikkan `./validate` pada console saat berada pada direktori NS.

NS dapat mensimulasikan jaringan TCP/IP dengan berbagai medianya. NS melaksanakan bermacam-macam protocol jaringan (TCP, UDP, RTP), Sumber trafik (FTP, Telnet, CBR), Mekanisme manajemen antrian pada router (DropTail, RED dan CBQ), algoritma routing Unicast (Distance Vector, Link State) dan Multicast (PIM SM,

PIM DM, DVMRP, Shared Tree dan Bi directional Shared Tree). NS juga mengimplementasikan beberapa MAC (IEE 802.3, 802.11) di berbagai media, misalnya jaringan *wired* (seperti LAN, WAN, point to point), *wireless* (seperti mobile IP, wireless LAN), bahkan simulasi hubungan antar-node jaringan yang menggunakan media satelit.

Network Simulator dibangun dengan menggunakan 2 bahasa pemrograman, yaitu C++ dan Tcl/OTcl. C++ digunakan untuk *library* yang berisi *event scheduler*, *protocol* dan *network component* yang diimplementasikan pada simulasi oleh *user*. Tcl/OTcl digunakan pada *script* simulasi yang ditulis oleh NS *user* dan pada *library* sebagai simulator *object*. Otcl juga nantinya akan berperan sebagai *interpreter*.

ns-2 [21, 22] adalah sebuah simulator kejadian diskrit yang dikembangkan oleh *University of California at Berkeley* dan projek *VINT* untuk keperluan penelitian dalam bidang jaringan (*networking*). *ns-2* menyediakan dukungan substansial untuk simulasi TCP, *routing*, protokol-protokol *multicast* pada jaringan kabel atau pun jaringan *wireless* termasuk juga satelit. *ns-2* memiliki keterbatasan karena hanya menggunakan bidang 2 dimensi dengan model refleksi pentanahan dua arah.



Gambar 3.1. Bagan kerja simulator *ns-2* [21]

Simulator ditulis dalam bahasa pemrograman C++, mendukung basis bahasa skrip OTCL pada Tcl/Tk. Para peneliti menentukan komponen-komponen jaringan seperti node-node, jalur / link, protokol-protokol dan trafik yang diperlukan menggunakan skrip OTCL. Simulator menggunakan skrip tersebut dan menghasilkan catatan pada lapisan-lapisan pilihan yang berbeda. Hasil-hasil simulasi digunakan untuk menghitung waktu tunda, *throughput*, konsumsi daya dan unjuk kerja pengukuran lainnya.

Penulis telah menggunakan *ns-2* untuk mensimulasikan *routing* protokol-protokol *ad hoc*. Dalam penelitian ini penulis menggunakan *ns-2* versi 2.28 yang dirilis bulan desember 2004. Pemodelan *wireless* pada *ns-2* terdiri dari sebuah node bergerak sebagai intinya. Model tersebut dapat ditambahi dengan kemampuan untuk mensimulasi jaringan *ad hoc* multihop, hingga menjadi *wireless* LAN. Sebuah node bergerak juga memiliki kemampuan untuk bergerak dalam topologi yang diberikan, kemampuan untuk mengirim dan menerima sinyal pada sebuah kanal *wireless*.

3.1.2. Pemodelan Lapisan Fisik

Model-model propagasi digunakan untuk menentukan jika data dikirimkan melalui udara berhasil diterima dengan baik. Model-model tersebut mempertimbangkan faktor-faktor waktu tunda propagasi, perabaan sinyal pembawa (*carrier sensing*), dan efek-efek penangkapan (*capture effects*).

Untuk pelemahan daya sinyal pada *ns-2* digunakan model ruang bebas $1/r^2$ untuk jarak pendek (r), dan model refleksi pentanahan $1/r^4$ untuk jarak jauh di atas

100 meter biasanya digunakan untuk antena dengan penguatan rendah, terletak dengan ketinggian 1,5 meter diatas permukaan tanah, pada band frekuensi operasi 1 – 2 GHz (r adalah jarak antar antena) [23, 24].

Ketika sebuah paket hendak dikirimkan, objek pengirim menghitung tunda waktu propagasi dari dirinya ke semua *interface* lainnya pada kanal, dan menjadwalkan kejadian penerimaan paket untuk tiap node. Penjelasan lebih jauh secara detil akan dibahas pada pergerakan node.

3.1.2.1. Implementasi Protokol MAC 802.11

Protokol MAC 802.11 diimplementasikan dalam *ns-2*. Lapisan MAC mengerjakan deteksi *collision*, fragmentasi dan *acknowledgement*. Protokol tersebut mungkin juga digunakan untuk mendeteksi kesalahan pengiriman. IEEE 802.11 adalah protokol CSMA/CA, yang akan menghindari terjadinya *collision* dengan terlebih dahulu memeriksa kanal sebelum menggunakannya. Jika kanal dalam keadaan bebas, ia akan mulai mengirimkan paket, jika kanal sedang digunakan node lain, MAC akan menunggu beberapa saat secara acak sebelum melakukan pengiriman ulang. Untuk setiap kali mencoba, digunakan algoritma *backoff* eksponensial. Pada medium udara, tidak bisa diasumsikan bahwa semua node stasiun mendengarkan satu sama lainnya. Jika sebuah stasiun menganggap bahwa media / kanal bisa digunakan, hal itu mungkin saja kanal tidak selamanya benar-benar bebas. Problem ini dikenal dengan problem stasiun tersembunyi, dan untuk mengatasi problem tersebut, mekanisme penghindaran *collision* dan skema *acknowledgement* positif digunakan bersama-sama. *Acknowledgement* positif

memerlukan node-node lain sekitarnya untuk mengirim ulang data dan menjawab satu sama lain sampai data berhasil dikirim ke tujuan.

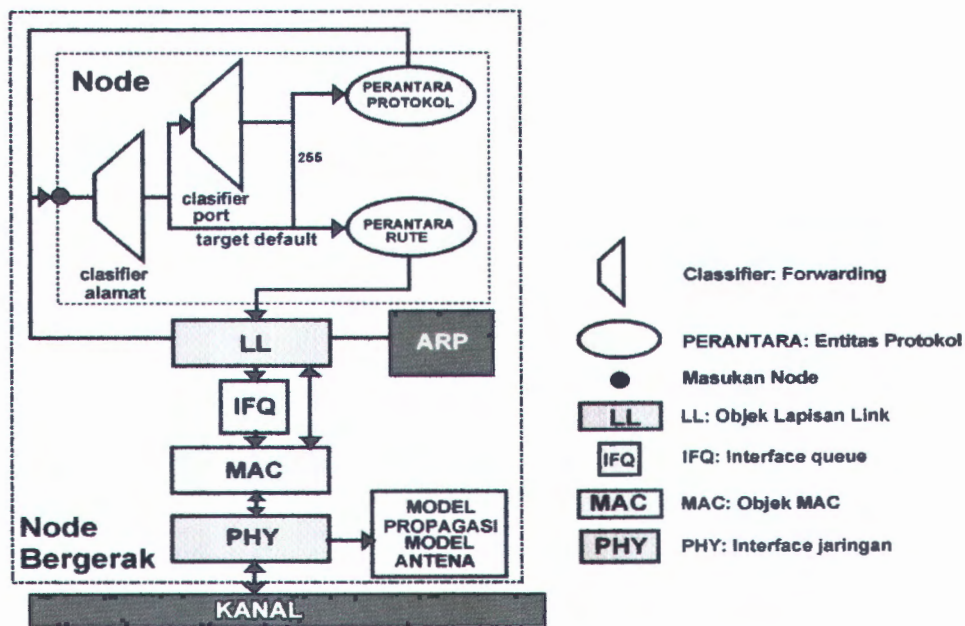
Address Resolution Protocol (ARP) juga diimplementasikan dalam *ns-2*. ARP mengubah alamat IP menjadi alamat *hardware* MAC sebelum paket-paket dikirimkan ke *hardware* melalui MAC. Penguatan antena dan parameter kepekaan penerima juga dapat ditentukan pada simulator *ns-2*. Ada beberapa antena yang bisa dipilih / ditentukan untuk proses simulasi. Implementasi penggunaan kanal berdasarkan pada model media bersama. Semua node-node bergerak memiliki satu atau lebih *interface* jaringan yang terhubung ke kanal. Sebuah kanal memiliki ketentuan frekuensi radio dengan keterangan modulasi dan sebuah skema pengkodean. Kanal-kanal adalah ortogonal, oleh sebab itu paket-paket yang dikirim pada satu kanal tidak berinterferensi dengan pengiriman dan penerimaan kanal yang berdekatan atau kanal-kanal yang lain. Sebuah paket akan diterima jika jangkauan pengiriman masuk dalam kalkulasi model propagasi radio, yang juga termasuk kesalahan bit (*bit error*) yang menyertainya.

3.1.3. Pergerakan Node

Node bergerak adalah objek dasar simulator *ns-2*, dengan penambahan fungsi-fungsinya, node dapat melakukan gerakan, menerima, dan mengirim data pada sebuah kanal. Kemampuan gerakan meliputi gerakan node, pembaruan informasi posisi secara periodik, dan pemeliharaan batas topologi. Kesemuanya diimplementasikan dalam bahasa pemrograman C++. Pembuatan dan penentuan komponen-komponen jaringan seperti penggolongan / *classifier*, demultiplekser, lapisan link (LL), MAC,

kanal pada sebuah node bergerak akan diimplementasikan dalam OTCL. Pembuatan dan penentuan node bergerak secara detil dijelaskan pada lampiran.

Tiap node bergerak ditentukan menjadi perantara *routing* (*routing agents*) untuk mengalkulasi *routing-routing* ke node-node lainnya dalam jaringan. Paket-paket yang dikirim dari aplikasi simulator diterima melalui perantara *routing* seperti pada gambar 3.2. Perantara *routing* menentukan sebuah jalur *routing* untuk tiap paket dan menandainya. Perantara *routing* kemudian mengirimkannya ke lapisan link. Lapisan link menggunakan protokol ARP untuk menentukan alamat *hardware* dari node-node tetangga, agar dapat memetakan alamat-alamat IP tersebut ke *interface-interface* mereka dengan benar. Kemudian paket-paket dikirimkan ke antrian *interface* (*IFq*), di situ paket menunggu sampai sebuah sinyal dari MAC diterima. Paket-paket kemudian meninggalkan antrian *interface* (*IFq*) menunggu sampai MAC mengirimkannya pada saat kanal dapat dipergunakan. Paket disalin ke semua *interface* jaringan pada saat bit pertama paket mulai sampai di *interface* seperti yang dilakukan pada sistem fisik nyata. Tiap *interface* jaringan menandai paket tersebut dengan cara masing-masing, dan memerlukan model propagasi. Sebagai catatan model propagasi diperlukan pada bagian penerima. Model propagasi menggunakan penandaan pengiriman dan penerimaan untuk menentukan daya pada saat *interface* akan menerima paket tersebut. *Interface* penerimaan jaringan ditinggalkan untuk kemudian memutuskan apakah paket diterima dengan baik atau tidak. Jika diterima dengan baik, paket diteruskan ke lapisan MAC. Jika lapisan MAC menerima paket tersebut sebagai paket bebas kesalahan dan bebas *collision*, MAC akan meneruskan ke titik masukan dari node.

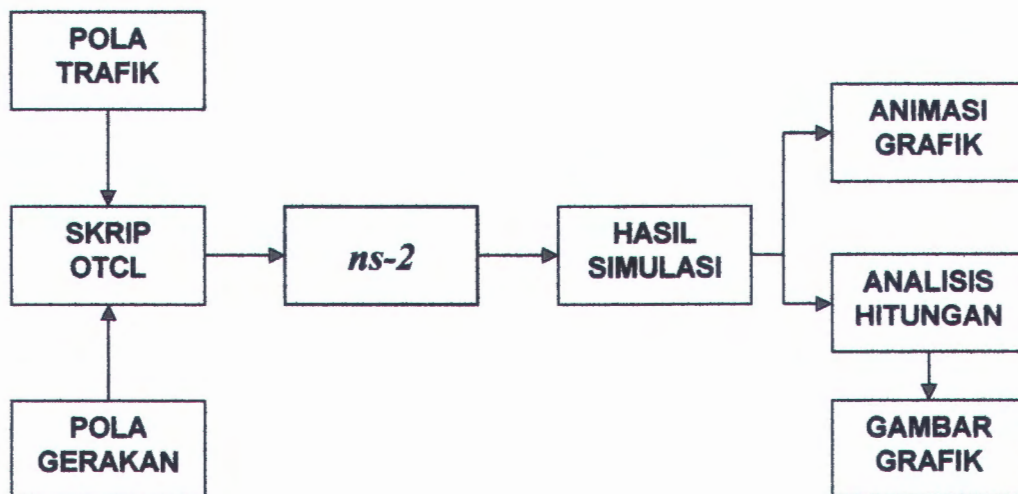


Gambar 3.2. Node bergerak (mobile node) *ns-2* [22]

Paket kemudian sampai pada masukan sebuah demultiplekser, yang akan menentukan apakah paket akan diteruskan ke node lain atau sudah sampai pada node tujuan. Jika titik kedatangan adalah node tujuan, paket akan dikirimkan ke demultiplekser, yang menentukan aplikasi simulasi, ke mana paket akan disampaikan. Jika paket diteruskan ke node lain, operasi di atas akan diulang [22].

3.2. Blok Diagram Simulasi

Gambar 3.3. menunjukkan blok diagram proses simulasi *ns-2* untuk simulasi *wireless ad hoc* yang penulis gunakan untuk melakukan simulasi. Mesin PC Pentium IV dengan memory 512 Mbyte, sistem operasi *Microsoft Windows* dan emulator unix *Cygwin* dengan system operasi Linux diinstal *ns-allinone-2.28* network simulator sebagai pusat pemroses simulasi.



Gambar 3.3. Blok diagram proses simulasi

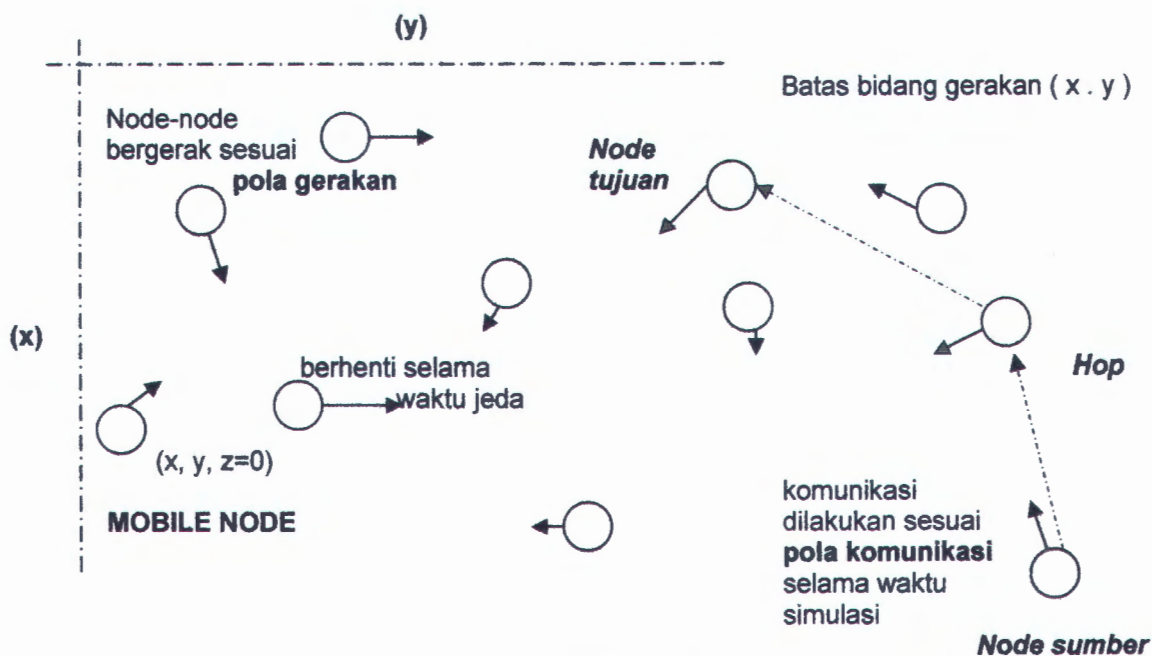
Skrip OTCL sebagai program utama simulasi yang mendiskripsikan: komponen jaringan seperti luas area gerakan node, kanal, propagasi, radio, antena, media fisik, protokol routing dan pengaturan waktu yang digunakan untuk proses simulasi dimasukkan ke mesin *ns-2*. Skrip program simulasi tersebut menggunakan file pola gerakan node dan file pola trafik komunikasi untuk disimulasikan oleh *ns-2*. Hasil proses simulasi ini kemudian bisa digunakan untuk menganalisis metrik-metrik unjuk kerja protokol routing yang lebih jauh akan dibahas pada bagian berikut, hingga dihasilkan grafik perilaku protokol yang dievaluasi. atau hasil simulasi tersebut dapat dianimasi menggunakan *Network Animator* (NAM) yang menjadi satu paket dengan *ns-2*, untuk penelitian menggunakan NAM (*Network Animator*).

3.2.1. Pola Gerakan Node dan Pola Trafik

Menggunakan utilitas yang sudah tersedia pada *ns-2* dapat disusun skenario pola gerakan node-node dalam area datar jaringan *ad hoc* yang sudah dipilih. Pola

gerakan node yang berbeda dibangkitkan menggunakan model *waypoint* [24,25]. Pada model *waypoint*, node terletak tersebar secara acak dalam area yang telah ditentukan batasnya. Tiap node bergerak menuju tujuan tertentu dalam area sesuai dengan kecepatan dan jeda sepanjang durasi simulasi yang ditentukan. Setelah jeda, node memilih tujuan lain dengan kecepatan lain melanjutkan proses simulasi. Parameter algoritma *waypoint*, kecuali seting waktu jeda didapatkan dari generator jumlah acak untuk distribusi yang ditetapkan. Untuk keperluan itu digunakan utilitas *setdest* yang tersedia di simulator *ns-2*. Contoh file pola gerakan node yang digunakan ada pada lampiran tesis ini.

File pola gerakan yang digunakan pada simulasi ini menggunakan karakter waktu jeda. Pada simulasi tiap node mulai bergerak berdasarkan waktu jeda. Node-node kemudian memilih arah gerakan secara acak pada bidang 1000 x 1000 meter persegi berjalan ke tujuannya dengan kecepatan tertentu, maksimum 20 meter per detik. Setelah sampai di tujuan ia akan berhenti selama waktu jeda, untuk kemudian melanjutkan gerakan berjalan menuju ke tujuan lain, berhenti lagi selama waktu jeda, dan seterusnya sampai habis waktu simulasi. Tiap simulasi dilakukan selama 500 detik. Dan simulasi dijalankan menggunakan pola gerakan node berpindah sebanyak 5 macam waktu jeda: 0, 40, 100, 200 dan 400 detik. Waktu jeda 0 detik menunjukkan bahwa node-node dalam jaringan selalu bergerak tidak berhenti, dan waktu jeda 400 detik menunjukkan node-node dalam jaringan bergerak amat lambat mendekati berhenti. Untuk meningkatkan akurasi analisa hasil simulasi dan kemungkinan kesalahan pembangkitan pola gerakan acak, pola gerakan tiap waktu jeda dibuat lebih dari satu file.



Gambar 3.4. Gambaran topologi gerakan Node Mobile dan koneksi komunikasi pada proses simulasi di mesin *ns-2*

Untuk mencapai tujuan simulasi yang dilakukan yakni mengevaluasi dan membandingkan unjuk kerja *routing* protokol-protokol *proactive wireless ad hoc*, dipilih sumber trafik yang konstan, *constant bit rate* (CBR). Simulasi ini menggunakan laju data 4 paket sebesar 512 byte perdetik. Tidak digunakan trafik TCP, karena TCP mengimplementasikan *time out* dan pengiriman ulang, juga melakukan pengaturan laju pengiriman data berhubungan dengan kondisi / kesibukan jaringan. Sehingga untuk menjaga kemungkinan waktu dan dari mana asal paket dikirimkan, node dan posisi pengirim, berbeda untuk protokol yang dievaluasi. Maka digunakan pola trafik yang sama untuk semua pola gerakan waktu jeda dan protokol yang dievaluasi. Untuk keperluan tersebut *ns-2* juga menyediakan pembangkit pola komunikasi [*cbrgen.tcl*] untuk menghasilkan pola trafik CBR dan TCP dengan menetapkan

jumlah node, laju data, ukuran paket dan jumlah koneksi (pasang) komunikasi data. File pola trafik komunikasi ada pada lampiran tesis ini. Akhirnya untuk menguji beban kerja protokol yang dievaluasi digunakan pola trafik komunikasi berbeda untuk tiap pembebanan kerja protokol yakni dengan cara mengatur jumlah hubungan komunikasi antar node dengan sumber CBR. Tabel 3.1. menunjukkan beban kerja yang diberikan saat simulasi.

Semua pola komunikasi adalah *peer to peer*, koneksi dimulai pada sebaran waktu serba sama antara 0 sampai 180 detik. Ketiga pola komunikasi (5, 10 dan 15 sumber) digabungkan dengan semua pola gerakan node untuk kemudian digunakan sebagai penguji protokol routing yang akan dibandingkan.

Lebih jauh proses pembangkitan pola koneksi trafik dan pola gerakan node.

Tabel 3.1. Definisi beban kerja untuk bermacam pola trafik

Beban Kerja	Sumber Trafik CBR
1	CBR/UDP 5 koneksi acak Panjang paket = 512 bytes
2	CBR/UDP 10 koneksi acak Panjang paket = 512 bytes
3	CBR/UDP 15 koneksi acak Panjang paket = 512 bytes

3.2.2. Routing Protokol Protokol Proactive yang dievaluasi

Simulasi yang penulis lakukan berdasarkan pada faktor-faktor spesifik *routing* protokol-protokol *proactive* yang dievaluasi sebagai berikut:

3.2.2.1. OLSR (Optimized Link State Routing)

Optimized Link State Routing (OLSR) protokol adalah seperti pada satu table-driven protokol. Dalam OLSR, node akan menukar pesan dengan node yang dekat lain dalam jaringan secara reguler untuk topologi informasi terbaru pada masing-masing node. Protokol OLSR menggunakan IP level pesan HELLO untuk mendeteksi kemungkinan link putus. Node akan menentukan *one-hop* ke tetangga mereka, yaitu. *Node* dengan radius dalam satu transmisi, dengan mulai pengiriman HALLO 3 detik. Rute balik memiliki waktu hidup lebih pendek, hanya 3 detik Kemudian pesan jawaban

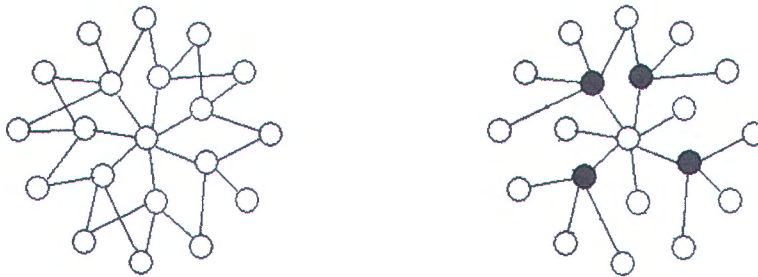
Tabel 3.2 Konstanta yang digunakan pada Simulasi OLSR

Parameter	Ukuran
<i>Interval</i> HELLO	2s
Waktu hidup rute	3s
Waktu hidup jawaban rute	5s
Mulai pengiriman	3.0 s
Delay (waktu tunda)	25 μ s



rute sebaiknya bisa diterima dalam waktu 5 detik setelah pengiriman. Usaha untuk pencarian rute hanya dicoba dengan melakukan sebanyak Suatu ukuran didasarkan pada pemilihan yang akan dijadikan dalam bagian berikut, satu set node di antara *one-hop* dengan tetangga digunakan multipoint Relays (MPR). Dalam suatu topology hanya satu node akan mengirimkan informasi, maka node yang lain menyediakan informasi parsial tentang jaringan, kemudian MPRS akan membentuk link informasi untuk dikirimkan ke seluruh jaringan. Dengan dua optimisasi ini , jumlah transmisi jadi diperkecil, dengan demikian untuk mengurangi exploitasi dibandingkan dengan hubungan Link State Routing Protocol. Masing-masing node akan menggunakan topologi ini, bersama dengan HALLO yang dikumpulkan, untuk menghitung rute yang optimal pada semua node dalam jaringan.

Dalam simulasi ini semua node akan meng *update* terus menerus selama simulasi berjalan dan node yang terdekat (satu hop) akan membentuk MPR (Multipoint Relay), sedangkan lebih dari 2-hop akan membentuk MPRS. Bila ada sebuah host/node yang tidak dapat mendengar pesan pembaruan, maka dapat diasumsikan bahwa ia merupakan host/node yang tidak terjangkau dari hop. Waktu pengumuman rute keseluruhan kebanyakan dapat ditentukan untuk satu detik. Gambar 3.5. menunjukkan flooding dengan menggunakan mekanisme multipoint relay (MPR). Sumber dari pesan menjadi di tengah-tengah node, dan masing-masing garis menandakan adanya indikasi suatu pesan sedang dikirimkan ke luar.



Gambar 3.5. flooding menggunakan mekanisme multipoint relay (MPR).[3]

Semua cabang node akan memilih suatu MPRS dengan bebas, menggunakan probabilitas berbeda atau serupa untuk memilih suatu MPRS yang minimal.

3.2.2.2. FSR (Fisheye State Routing)

Dalam suatu lingkungan wireless, suatu radio menghubungkan antara node yang bergerak sering mengalami putus dan menyambung kembali. LS (Link State) protokol memberikan suatu link state yang baru untuk masing-masing perubahan seperti, *flood* ke jaringan dan menyebabkan eksploitasi yang berlebihan. Untuk menghindari masalah ini FSR menggunakan kelengkapan parameter wireless dan konstanta yang digunakan untuk simulasi FSR,

Tabel 3.3 Konstanta yang digunakan pada Simulasi FSR

Parameter	Nilai
Sumber pengiriman pertama	10s
Sumber pengiriman kedua	27s
Capture threshold	10,0 dB
Carrier sense threshold	$1,559 \times 10^{-11}$ W
Receive power threshold	$3,652 \times 10^{-10}$ W
Laju data	2 Mbps
Daya pancar	0,2818 W
Frekuensi	914×10^6 Hertz

Parameter setup simulasi untuk protokol rute FSR menggunakan pengiriman dengan dua sumber yaitu sumber pertama dimulai pada detik ke 10, sedangkan sumber kedua mulai mengirim pada detik ke 27. Dalam simulasi ini terdapat perbedaan antar routing OLSR dan FSR, simulasi FSR menggunakan Vector loss.

Inisialisasi parameter *interface* media bersama digunakan agar dapat bekerja seperti *interface* perangkat radio 914 MHz. Model propagasi dan daya pancar menentukan daya sinyal yang diterima dari tiap paket. Paket tidak akan dapat diterima dengan baik bila daya penerimaannya dibawah ambang batas (*threshold*). Keseluruhan table topologi ditukar dengan tetangga untuk memberikan suatu lokasi dengan bidang yang lebih luas dengan bandwidth yang besar, melalui link state informasi dengan frekuensi yang berbeda-beda tergantung pada jarak lingkup fisheye. FSR baik digunakan pada ukuran jaringan besar dan mengirimkan pesan kepada node terdekat. Dengan bantuan suatu rute sehingga dapat dikirim ke suatu tujuan guna untuk mengindari terjadinya hidden. Untuk meningkatkan mobility, yang mengarahkan pada rute tujuan akan menjadi akurat. Bagaimanapun, ketika sebuah paket mendekati tujuannya, menemukan peningkatan tentang akurasi instruksi rute yang masuk ke semua sektor.

3.2.3. Evaluasi Perbandingan Unjuk Kerja Routing Protokol jaringan Wireless Ad Hoc

Evaluasi unjuk kerja routing protokol-protokol proactive pada jaringan *wireless ad hoc* multi hop yang akan digunakan untuk perbandingan unjuk kerjanya dibahas berikut ini:

3.2.3.1. Keberhasilan Penyampaian Paket

Keberhasilan penyampaian paket, dalam persen adalah angka pecahan dari paket-paket yang sukses diterima setelah melalui perjalanan mencari node tujuan. Unjuk kerja ini mengukur dan menentukan kelengkapan dan kebenaran pengiriman paket yang dilakukan oleh routing protokol. Sukses pengiriman paket dihitung sebagai berikut, semua paket data dengan pengidentitas unik yang meninggalkan MAC sumber diperhitungkan dan ditentukan sebagai paket-paket asli. Paket yang diterima kemudian dibandingkan dengan database kumpulan pengiriman dan tiap paket unik dihitung sekali untuk meyakinkan pencegahan penghitungan lebih paket yang diterima disebabkan utamanya karena efek lintasan jamak akibat gerakan node. Hasil rata-rata rasio paket unik yang diterima dengan paket unik yang dikirim dirumuskan seperti persamaan 3.1.

$$F = \frac{1}{C} \sum_{f=1}^C \frac{R_f}{T_f} \quad (3.1)$$

F = Angka pecahan dari paket yang sukses disampaikan

C = Jumlah total aliran, koneksi

f = ID unik paket yang mengalir

R_f = Hitungan paket unik yang diterima dari aliran f

T_f = Hitungan paket yang dikirimkan ke aliran f

3.2.3.3. Throughput

Throughput adalah laju data yang sukses dikirimkan per detik pada jaringan selama simulasi berlangsung. *Throughput* dihitung sebagai jumlah ukuran / nilai muatan

yang berhasil dikirimkan dari paket-paket data dalam perioda, dimulai saat node sumber membuka pintu komunikasi menuju ke pintu node tujuan yang jauh, dan diakhiri saat simulasi dihentikan. *Header-header* routing OLSR dan FSR, pesan-pesan kontrol, pesan-pesan MAC dan semua bentuk pengiriman ulang diabaikan untuk mengkalkulasi rata-rata efektif *end-to-end throughput* untuk sebuah link komunikasi tunggal. *Throughput* dapat dihitung dengan persamaan 3.3.

$$\text{Throughput } \gamma = \frac{1}{C} \sum_{f=1}^C \frac{1}{(t_{end,f} - t_{start,f})} \sum_{i_f=1}^{T_f} (s_{i_f} - h_{i_f}) \quad (3.2)$$

γ = Throughput dalam bytes per detik

t_{end} = Waktu akhir pengiriman dalam detik

t_{start} = Waktu mulai pengiriman dalam detik

C = Jumlah total aliran dalam jaringan

f = ID unik aliran

i_f = ID unik paket dari aliran f

s_{i_f} = Nilai paket yang sukses dikirimkan dalam bytes

h_{i_f} = Nilai paket header yang sukses dikirimkan dalam bytes

3.2.4. Faktor-Faktor Simulasi

Ukuran dan bentuk tempat yang digunakan saat simulasi: Faktor ini terutama ukuran dapat berhubungan dengan kepadatan node. Jika area berbentuk segi empat akan lebih memungkinkan untuk dapat melakukan perjalanan paket dengan rutenya yang panjang. Simulasi yang dilakukan menggunakan ukuran tempat 1000 x 1000 meter

persegi, penggunaan bentuk segi empat bertujuan agar dapat mendapatkan rute-rute yang lebih panjang.

3.2.4.1. Model gerakan

Model gerakan yang digunakan adalah model *random waypoint* pada lapangan segi empat. Konfigurasi lapangan yang digunakan adalah 1000 x 1000 meter persegi berisi 100 node bergerak. Tiap paket memulai perjalanannya dari suatu node di tempat acak ke node tujuan yang juga acak dalam area simulasi dengan sebuah kecepatan acak yang dipilih (*uniform distributed*) antara 0 sampai 20 meter per detik. Sekali tujuan tercapai, tujuan acak lain ditargetkan setelah jeda (*pause*). Waktu jeda yang mempengaruhi kecepatan relatif dari gerakan node divariasikan selama waktu simulasi yakni: 0, 40, 100, 200, 400 detik. Simulasi dijalankan selama 500 detik. Gerakan identik dan skenario trafik digunakan untuk semua protokol yang dibandingkan untuk mendapatkan hasil yang lebih bagus.

3.2.5. Set-up Simulasi

Simulasi untuk menguji dan mengamati *routing* protokol *proactive* OLSR dan FSR dilakukan dalam dua macam kondisi. Kondisi pertama untuk meneliti dan mengamati akibat dari adanya gerakan node dengan bermacam kecepatan dalam jaringan. Simulasi kondisi kedua untuk meneliti dan mengamati akibat dari perubahan beban kerja jaringan. Tabel 3.4. menunjukkan set-up parameter untuk simulasi yang dilakukan.

Tabel 3.4. Set-Up Parameter Simulasi

Parameter	Ukuran
Jumlah node	100
Waktu jeda <i>Waypoint</i>	0, 40, 100, 200, 400 detik
Ukuran lapangan	1000 x 1000 meter persegi
Waktu simulasi	500 detik
Model propagasi	<i>Two ray ground</i>
MAC	802.11
<i>Bandwidth</i> pancaran	2 Mbps
Tipe antena	Antena Omni Directional
<i>Energy transmit power</i>	0,660 joule
<i>Energy receive power</i>	0,395 joule
<i>Energy idle power</i>	0,035 joule
Ukuran panjang paket UDP maksimal	1460 bytes
Tipe antrian	<i>Droptail priority</i>

BAB IV

HASIL DAN PEMBAHASAN

Bab ini memaparkan hasil pengamatan simulasi unjuk kerja yang dilaksanakan menggunakan acuan yang telah ditentukan pada BAB III.

4.1 Gerakan Node

Sub bab ini menampilkan hasil simulasi akibat yang ditimbulkan karena perubahan kecepatan gerak dari node-node jaringan pada unjuk kerja protokol-protokol rute yang dianalisa seperti tersebut pada bagian 3.3.1. Waktu jeda 0 detik menunjukkan bahwa node-node dalam jaringan selalu bergerak tidak berhenti, dan waktu jeda 400 detik menunjukkan node-node dalam jaringan bergerak amat lambat mendekati berhenti. Oleh sebab itu, pengamatan dititikberatkan perubahan nilai sumbu Y terhadap sumbu X (waktu jeda \cong kecepatan gerakan node) gambar grafik yang ditampilkan dari tabel hasil simulasi. Protokol rute dengan simpangan maksimum dari nilai rata-rata = kecil, menunjukkan bahwa protokol tersebut cepat beradaptasi atau cepat konvergensi terhadap perubahan topologi akibat kecepatan gerakan node dalam jaringan.

Dengan meningkatkan rata-rata kecepatan gerakan node, akan meningkat pula laju kejadian kesalahan rute. Hal tersebut terjadi karena kemungkinan untuk membuang (*drop*) paket yang ada pada perantara antrian akan dilakukan protokol, karena kondisi sulitnya menemukan hop berikut rute ke node tujuan yang telah ditentukan untuk waktu yang terbatas.

Analisa perilaku protokol rute yang akan dibandingkan dilihat dari:

- keberhasilan penyampaian paket.
- tunda waktu yang terjadi saat penyampaian paket dari node sumber ke tujuan.
- *throughput* yang dihasilkan pada node penerima.
- beban kontrol rute karena perubahan kecepatan gerakan node dalam jaringan.

Rangkuman perilaku akibat gerakan node akan menggambarkan perilaku protokol-protokol yang dianalisa.

4.1.1. Keberhasilan Penyampaian Paket

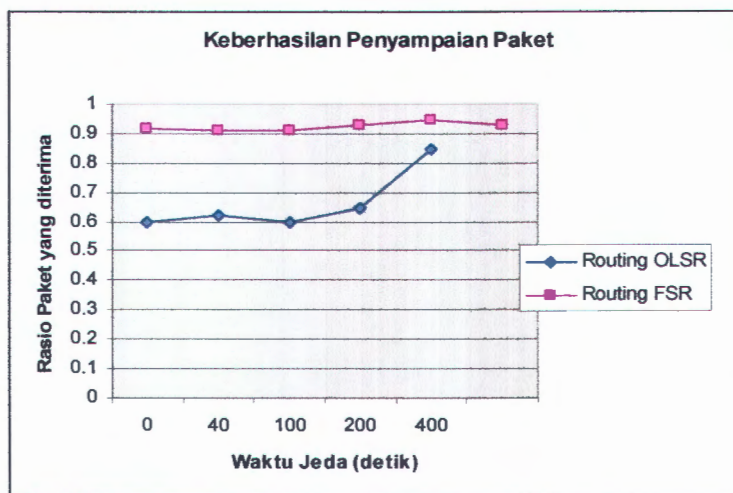
Gambar 4.1. menunjukkan grafik *packet delivery ratio* dari tabel 4.1., perbandingan antara paket yang dikirimkan dan paket yang diterima rata-rata node-node dalam jaringan, atau disebut juga metrik keberhasilan penyampaian paket, dibandingkan dengan bermacam waktu jeda. Simulasi dilakukan dengan pola gerakan untuk 5 waktu jeda yang berbeda: 0, 40, 100, 200 dan 400 detik. Waktu jeda 0 detik dapat disamakan dengan melanjutkan gerakan pada kecepatan acak, maksimum 20 meter perdetik. Dan waktu jeda 400 detik mendekati waktu berhenti proses simulasi 500 detik, atau dapat disamakan dengan tanpa gerakan node.

Protokol OLSR memiliki sifat pasif menunggu jawaban *acknowledgement* untuk mendeteksi kegagalan link dan pesan kesalahan rute yang menginformasikan ke sumber terjadi invalidasi rute. Ketika laju *collision* meningkat, pesan kontrol spesial sulit untuk dilakukan, akibatnya mengurangi kemampuannya untuk menemukan rute baru. OLSR mampu menyampaikan paket ke tujuan dengan baik rata-rata sekitar 96,6% (minimal 95,6% dan maksimal 98,4%). Pada gerakan node kecepatan tinggi, jeda antara 0 – 200 detik, protokol ini memiliki kinerja

penyampaian paket lebih buruk, yakni hanya menyampaikan paket sekitar 95%-nya saja. Penyebab keadaan ini karena kegagalan operasi penerima saat kekurangan informasi komunikasi *acknowledgement*. Meskipun demikian, diantara protokol yang dipelajari, OLSR memiliki kemampuan penyampaian paket terbaik untuk kondisi akibat perubahan kecepatan gerakan node.

Tabel 4.1. Rasio keberhasilan penyampaian paket

Keberhasilan Penyampaian Paket		
wkt.jeda p.rute	OLSR	FSR
0	0.6	0.917
40	0.62	0.909
100	0.599	0.91
200	0.643	0.927
400	0.844	0.946
rata-rata	0.702	0.928



Gambar 4.1. Keberhasilan penyampaian paket fungsi waktu jeda (*Packet Delivery Ratio*)

Protokol OLSR gagal untuk mencapai konvergensi dibawah waktu jeda 200 detik, saat ia mampu menyampaikan 85 % paket yang dikirim. Pada gerakan node kecepatan tinggi, waktu jeda anta 0 – 200 detik, OLSR mempunyai kinerja buruk. Ia hanya mampu menyampaikan paket sekitar 60% saja. 40% paket yang dibuang

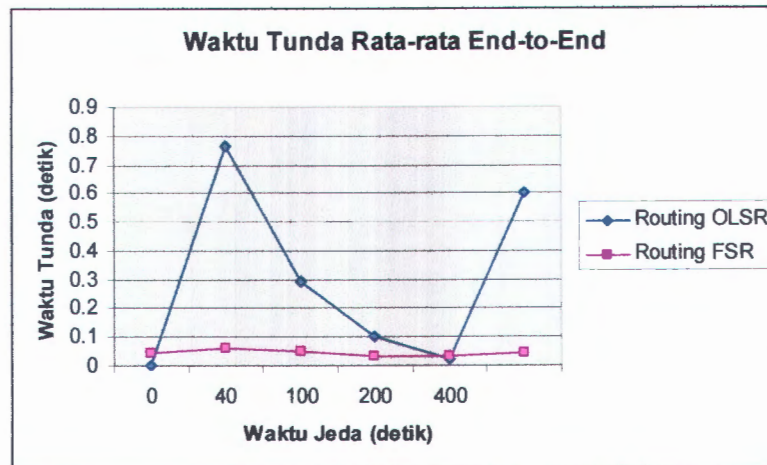
hilang karena tabel rute yang usang mengarahkan paket menuju ke tujuan banyak menemui link

4.1.2. Tunda Waktu antara Node Sumber ke Node Tujuan

Terdapat beberapa faktor yang menyebabkan kenaikan waktu tunda penyampaian paket ke tujuan. Pencarian rute menyebabkan protokol rute memasukkan paket-paket yang harus diteruskan, masuk dalam antrian untuk menunggu selama waktu pencarian rute oleh protokol. Waktu tunda penyampaian paket juga naik jika ia diteruskan melalui lebih banyak hop oleh mekanisme dan kemampuan protokol rute untuk mendapatkan lintasan dengan hop minimum. Pengukuran waktu tunda node sumber ke node tujuan ini dilakukan hanya pada paket yang dikirimkan dari sumber dengan identitas level penelusuran tertentu dan diterima di node tujuan untuk paket yang beridentitas sama. Gambar 4.2. menunjukkan grafik tunda waktu antara node sumber ke node tujuan rata-rata dibandingkan dengan waktu jeda yang identik berlawanan dengan kecepatan gerakan node dalam jaringan hasil simulasi yang dilakukan.

Tabel 4.2. Waktu tunda penyampaian paket

Waktu Tunda Rata-rata (detik)		
wkt.jeda p.rute	OLSR	FSR
0	1.458	0.044
40	0.763	0.063
100	0.294	0.05
200	0.104	0.033
400	0.023	0.032
rata-rata	0.603	0.046



Gambar 4.2. Waktu tunda antara node sumber ke node tujuan rata-rata fungsi waktu jeda

Protokol rute yang mempunyai rata-rata tunda waktu terkecil untuk menyampaikan paket dari node sumber ke node tujuan dalam hubungannya dengan kecepatan gerakan node pada jaringan *ad hoc* adalah FSR, waktu tunda rata-rata yang diperlukan protokol ini sekitar 46 *milidetik*. Perubahan kecepatan gerakan node yang ditunjukkan dengan waktu jeda memberi pengaruh yang tidak signifikan. Hal ini dapat dijelaskan sebagai berikut: bahwa unjuk kerja FSR yang baik dalam menjaga paket kontrol rute, dengan sedikit laju membuang paket yang ada pada perantara antrian (*Ifq*). Buf er perantara antrian yang diset 64 paket cukup untuk menempatkan paket menunggu pencarian rute menuju ke node tujuan, sehingga hanya dilakukan pembuangan paket sedikit, akhirnya tunda waktu penyampaian paket juga kecil.

Protokol OLSR dan FSR, pada saat node bergerak cepat (jeda waktu antara 0 – 100 detik) menyampaikan paket ke tujuan dengan tunda waktu rata-rata yang lebih besar dari 200 *milidetik*. Pada umumnya protokol rute pada jaringan *ad hoc* dengan node yang bergerak cepat, akan menunda penyampaian paket ke tujuan karena

mereka memerlukan waktu untuk menemukan rute yang benar sesuai mekanisme dan algoritma yang digunakan agar dapat menyampaikan paket. Paket-paket yang akan dikirimkan diletakkan pada perantara antrian saat protokol bekerja menemukan rute penyampaian paket ke tujuan.

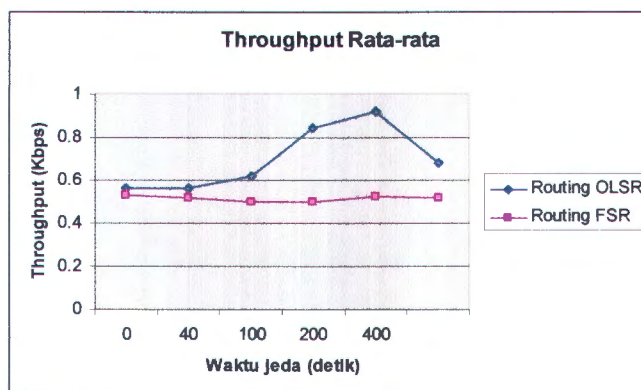
4.1.3. Throughput antara Node Sumber ke Node Tujuan

Gambar 4.3. menunjukkan *throughput* rata-rata masing-masing protokol rute yang diukur berdasarkan waktu jeda sebagai kecepatan gerakan node. *Throughput* dihitung sebagai jumlah ukuran muatan yang berhasil dikirimkan dari paket-paket data dalam perioda per detik, dimulai saat node sumber membuka pintu komunikasi menuju ke pintu node tujuan yang jauh, dan diakhiri saat simulasi dihentikan.

Dari gambar 4.3. terlihat bahwa grafik *throughput* protokol

Tabel 4.3. Throughput Rata-rata

Throughput Rata-rata		
Wkt.jeda p.rute	OLSR	FSR
0	0.561	0.531
40	0.564	0.517
100	0.62	0.5
200	0.845	0.503
400	0.917	0.524
rata-rata	0.681	0.517



Gambar 4.3. Throughput rata-rata antara node sumber ke node tujuan fungsi waktu jeda

Grafik *throughput* yang dihasilkan oleh protokol OLSR menunjukkan bahwa perilakunya simetri dengan keberhasilan penyampaian paket yang dilakukan oleh protokol ini. Setelah masalah pembaruan tabel rute dilakukan, paket yang dibuang berkurang sehingga *throughput* yang dihasilkan meningkat, sebanding dengan melemahnya gerakan node. Hasil pengukuran *throughput* yang didapat menggunakan protokol OLSR rata-rata adalah *0,68 kbps*, nilai minimalnya adalah *0,56 kbps* dan maksimalnya *0,92 kbps*.

Throughput yang lebih stabil dihasilkan oleh protokol FSR, kondisi simulasi dengan model gerakan node dan pola koneksi yang dipergunakan kemungkinan besar mendukung mekanisme kerja protokol ini. Perilaku FSR keberhasilan penyampaian paket dan tunda waktu penyampaian paket menunjukkan kestabilan dari protokol ini. Hasil pengukuran *throughput* yang didapat menggunakan protokol FSR rata-rata adalah *0,52 kbps*, nilai minimalnya adalah *0,50 kbps* dan maksimalnya *0,53 kbps*.

Pada umumnya *throughput* yang dihasilkan oleh protokol rute yang berdasarkan inisialisasi sumber *Link State* pada simulasi yang dilakukan mempunyai kecenderungan menurun mulai dari start, sebanding dengan penurunan kecepatan gerakan node. Sebaliknya untuk protokol yang menggunakan dasar kerja dikemudikan oleh tabel naik mulai start dan berbanding terbalik terhadap kecepatan gerakan node dalam jaringan.

4.1.5. Rangkuman Akibat Gerakan Node

Meningkatnya perubahan jaringan karena perubahan kecepatan gerakan node-node yang ada pada jaringan *ad hoc* menyebabkan menurunnya kemampuan keberhasilan penyampaian paket ke node tujuan. Utamanya untuk protokol-protokol

dengan mekanisme kerja berdasarkan tabel, pengiriman pesan kontrol juga meningkat saat diperlukan dihadapkan pada masalah link yang terputus. Protokol-protokol yang melakukan pembaruan secara periodik mengalami keterlambatan informasi rute untuk mengadaptasi perubahan topologi jaringan.

Peningkatan perubahan jaringan karena perubahan kecepatan gerakan node-node juga menyebabkan terjadinya peningkatan tunda waktu penyampaian paket data. Hal ini karena paket terlalu lama menunggu pada bufer perantara antrian menanti router yang sibuk menemukan jalur rute tepat untuk menuju ke node tujuan. Protokol pesan kontrol *collision* dan pesan kontrol protokol MAC 802.11 meningkatkan durasi waktu untuk menunggu pada antrian kepada tiap paket yang hendak dikirimkan. Jika waktu maksimum yang ditentukan oleh protokol rute terlewati, maka paket-paket dalam antrian akan dibuang. Hal ini dilakukan untuk mengurangi tunda waktu jaringan, sementara laju keberhasilan penyampaian paket juga akan menurun pada perubahan jaringan cepat.

Gerakan node pada jaringan memiliki akibat kecil terhadap simpangan rata-rata hop, karena hal itu sangat tergantung pada protokol rute, topologi dan jumlah node yang ada pada jaringan.

4.2. Akibat Kenaikan Beban Kerja

Perubahan beban kerja jaringan dilakukan dengan meningkatkan jumlah koneksi antar node dengan sumber CBR, 5, 10 dan 15 koneksi maksimum. Dengan cara meningkatkan jumlah koneksi komunikasi antar node pada jaringan, diasumsikan akan mengakibatkan terjadinya peningkatan jumlah rute-rute komunikasi dan lebih banyak host / node yang aktif melakukan dukungan

komunikasi pada proses rute. Hal tersebut akan memungkinkan terjadinya peningkatan jumlah paket yang hilang dan paket yang dibuang. nilai rata-rata yang terkecil menunjukkan unjuk kerja terbaik protokol menghadapi perubahan beban kerja jaringan.

Pola gerakan 100 node-node dalam jaringan masih menggunakan kecepatan maksimum 20 meter perdetik waktu simulasi 500 detik dan enam buah waktu jeda: 0, 40, 100, 200, dan 400 detik.

4.2.1. Keberhasilan Penyampaian Paket

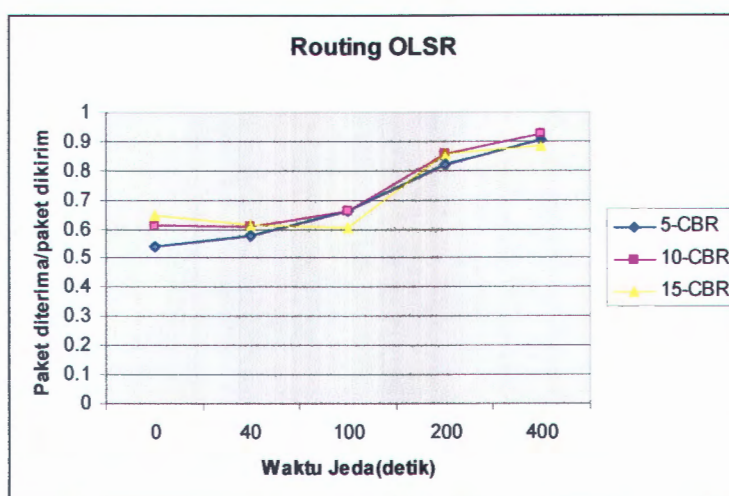
Gambar 4.4. menunjukkan grafik protokol-protokol rute dalam rasio keberhasilannya menyampaikan paket data untuk 3 jenis pola skenario komunikasi 5, 10 dan 15 koneksi trafik CBR dibandingkan dengan waktu jeda.

OLSR, gambar 4.4. menunjukkan perilaku yang tidak berbeda jauh saat terjadi peningkatan beban kerja jaringan. Beban kerja yang diwakili dengan perbedaan jumlah koneksi yang terjadi pada jaringan pengaruhnya tidak signifikan. Pada saat gerakan node cepat sampai kecepatan saat waktu tunda 200 detik keberhasilan penyampaian paket berkisar 60% dan menjadi sekitar 80% pada jeda 200 detik. Selanjutnya saat node mendekati diam keberhasilan penyampaian paket meningkat sampai sekitar 90%. Perilaku yang hampir sama dalam menghadapi perbedaan beban kerja jaringan disebabkan karena OLSR merupakan protokol rute yang kerjanya berdasarkan kebenaran tabel rute. OLSR juga merupakan suatu protokol yang menyampaikan paket ke node tujuan hop demi hop dalam vektor jarak, ia memerlukan node-node dalam jaringan secara periodik mengirimkan pesan pembaruan rute. Setelah masalah pembaruan rute terpenuhi, yakni tabel rute bisa

dijaga kebenarannya, maka untuk koneksi antar node antara 5 sampai 15 koneksi maksimum selama waktu simulasi 500 detik akan disampaikan dengan keberhasilan yang sama.

Tabel 4.4. Rasio keberhasilan penyampaian paket OLSR (pecahan/prosen)

OLSR				
wkt.jeda	p.rute	5-CBR	10-CBR	15-CBR
0		0.541	0.614	0.646
40		0.575	0.61	0.613
100		0.664	0.662	0.603
200		0.819	0.856	0.857
400		0.904	0.926	0.885
Rata-rata				0.702



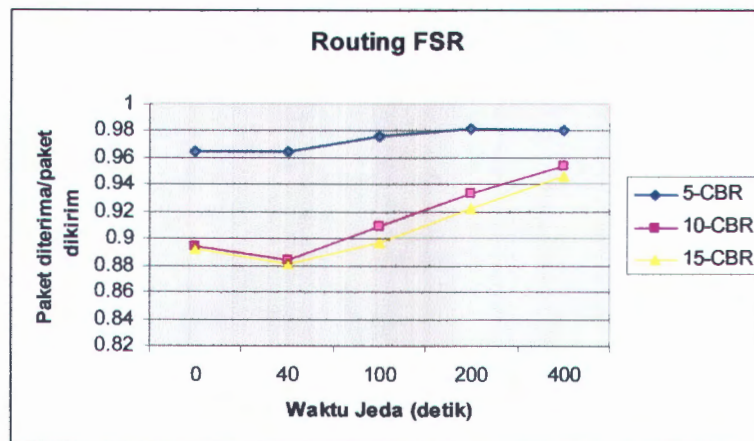
Gambar 4.4. Keberhasilan penyampaian paket pada protokol rute OLSR

Prototokol rute FSR gambar 4.4 menunjukkan perilaku penurunan keberhasilan penyampaian paket saat beban kerja jaringan meningkat. Protokol FSR memiliki perilaku keberhasilan penyampaian paket lebih baik saat ada 5 koneksi sumber CBR, antara 96,3% sampai 98,2%. Pada beban jaringan meningkat menjadi 10 koneksi maksimum sumber CBR, nilai keberhasilan penyampaian paket turun menjadi antara 88,4% sampai 95,4%. Dan antara 88,1% sampai 94,6% untuk beban

jaringan maksimum 15 koneksi sumber CBR. Penurunan keberhasilan penyampaian paket karena meningkatnya beban kerja jaringan bagi protokol FSR ini disebabkan karena semakin banyak koneksi terjadi mekanisme pencarian rute akan sering terjadi. Spasi *bandwidth* jaringan akan banyak digunakan untuk mekanisme pengiriman. Meskipun tidak terlalu banyak terjadi penurunan, yakni minimal antara 88% sampai 96%, akan tetapi untuk ukuran keberhasilan penyampaian paket, terlihat sangat dipengaruhi oleh kondisi beban jaringan.

Tabel 4.5. Rasio keberhasilan penyampaian paket FSR (pecahan/prosen)

FSR			
wkt.jeda p.rute	5-CBR	10-CBR	15-CBR
0	0.964	0.895	0.892
40	0.964	0.884	0.881
100	0.976	0.909	0.897
200	0.982	0.933	0.922
400	0.981	0.954	0.946
Rata-rata			0.928



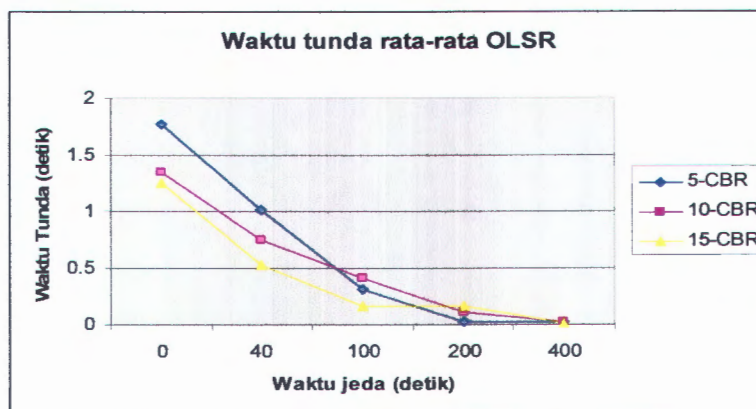
Gambar 4.5. Keberhasilan penyampaian paket rute protokol FSR

4.2.2. Waktu Tunda antara Node Sumber ke Node Tujuan

Gambar 4.6. menunjukkan tunda waktu yang dilakukan oleh protokol-protokol rute dalam menyampaikan paket ke tujuan berkaitan dengan peningkatan beban kerja jaringan dibandingkan dengan waktu jeda.

Tabel 4.6. Waktu tunda rata-rata (detik)

OLSR			
wkt.jeda p.rute	5-CBR	10-CBR	15-CBR
0	1.774	1.344	1.256
40	1.009	0.756	0.524
100	0.309	0.414	0.16
200	0.029	0.116	0.166
400	0.025	0.028	0.017
Rata-rata			0.603



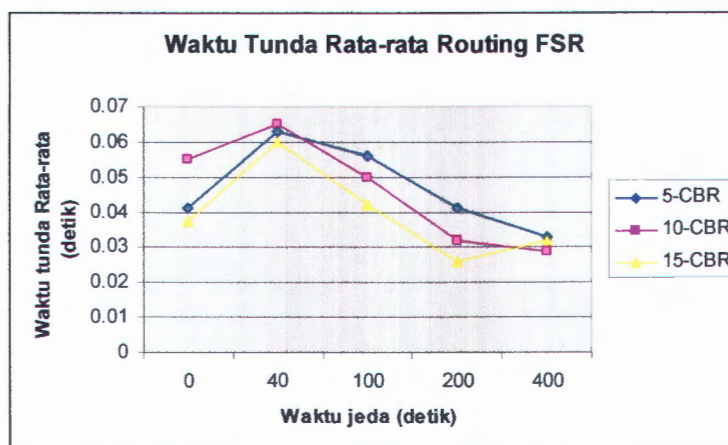
Gambar 4.6. Waktu tunda End-to-end rute protokol OLSR

Gambar 4.6 menunjukkan perbedaan tunda waktu protokol OLSR akibat meningkatnya beban kerja jaringan. Berbeda sedikit dengan perilakunya menyampaikan paket, pada beban kerja yang ringan terlihat saat kecepatan gerakan node cepat, tunda waktu lebih tinggi dibandingkan dengan saat beban kerja meningkat. Dari perilaku ini dapat disimpulkan bahwa semakin banyak node yang berinteraksi melakukan hubungan komunikasi, akan lebih sering terjadi pembaruan tabel rute. Karena saat sebuah node mulai mencari rute ia akan menyebarkan

permintaan ke alamat node tujuan, dan menghitung jumlah hop yang akan dilalui sampai ke node tujuan untuk dimasukkan ke tabel rutenya. Tiap node dalam jaringan yang mendengar interaksi ini akan membandingkan informasi yang diterimanya dengan tabel rute yang ia miliki. Jika informasi hop lebih baru dari tabel rutenya, ia akan memperbaiki susunan tabel rutenya berdasarkan metrik vektor jarak. Semakin sering dan semakin baru informasi yang diterima secara periodik, tabel rute yang dimilikinya semakin dapat dipercaya.

Tabel 4.7. Waktu tunda rata-rata (detik)

FSR				
wkt.jeda p.rute	5-CBR	10-CBR	15-CBR	
0	0.041	0.055	0.037	
40	0.063	0.065	0.06	
100	0.056	0.05	0.042	
200	0.041	0.032	0.026	
400	0.033	0.029	0.032	
Rata-rata			0.046	



Gambar 4.7. Waktu tunda End-to-end rute protokol FSR

Gambar 4.7 menunjukkan grafik perilaku tunda waktu dalam menyampaikan paket untuk protokol FSR menghadapi perubahan beban kerja jaringan. Secara

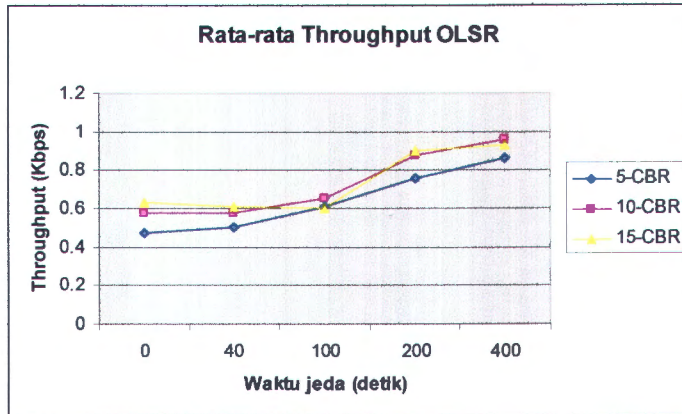
umum dapat ditinjau bahwa akibat kenaikan beban kerja tunda waktu yang terjadi dalam penyampaian paket juga semakin rendah. Karena topologi FSR membentuk terpusat dimana node-node yang terdekat dengan pusat dia akan leh kuat untuk memnerima pesan dalam pengiriman paket terlihat bahwa perilakunya pun hampir sama dengan perotokol pembentuknya. Semakin sering terjadi koneksi komunikasi data, maka upaya perawatan rute protokol ini akan semakin terbantu. Karenanya akan semakin singkat waktu yang diperlukan untuk menyampaikan paket dari node sumber ke node tujuan.

4.2.3. Throughput antara Node Sumber ke Node Tujuan

Gambar 4.8 menunjukkan grafik *throughput* protokol-protokol yang dipelajari dalam menghadapi peningkatan beban kerja jaringan. Peningkatan beban kerja jaringan dilakukan dengan menaikkan koneksi maksimum sumber CBR: 5, 10 dan 15 koneksi. Gambar 4.8 menunjukkan perilaku *throughput* yang dihasilkan protokol OLSR akibat perubahan beban kerja jaringan. Kenaikan beban kerja jaringan juga meningkatkan *throughput* yang dihasilkan node tujuan / penerima meskipun kurang signifikan.

Tabel 4.8. Rata-rata Throughput (kbps)

OLSR			
wkt.jeda p.rute	5-CBR	10-CBR	15-CBR
0	0.47	0.58	0.63
40	0.5	0.58	0.61
100	0.61	0.65	0.6
200	0.76	0.88	0.9
400	0.86	0.96	0.93
Rata-rata			0.68

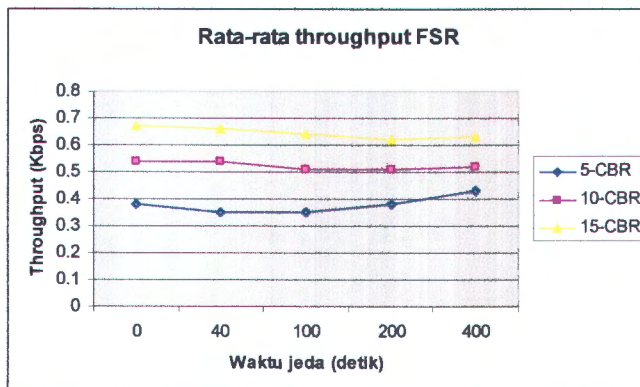


Gambar 4.8. Throughput rata-rata protokol OLSR

Hal yang sama juga ditunjukkan untuk perilaku keberhasilan penyampaian paket dan tunda waktu penyampaian paket ke node tujuan yang telah dibahas sebelum ini. Akibat masalah pencarian rute yang dihadapi saat kecepatan gerakan node tinggi, *throughput* yang dihasilkan juga rendah sekitar 500 bps.

Tabel 4.9. Rata-rata Throughput (kbps)

FSR				
wkt.jeda p.rute	5-CBR	10-CBR	15-CBR	
0	0.38	0.54	0.67	
40	0.35	0.54	0.66	
100	0.35	0.51	0.64	
200	0.38	0.51	0.62	
400	0.43	0.52	0.63	
Rata-rata				0.52



Gambar 4.9. Throughput rata-rata protokol OLSR

Karena sering terjadinya kesalahan pengiriman paket melalui link yang putus akibat tabel rute yang usang. Setelah problem pencarian rute mereda dengan melemahnya gerakan node *throughput* meningkat sampai maksimum 960 bps.

Gambar 4.9 menunjukkan grafik perilaku produksi *throughput* node penerima protokol FSR menghadapi peningkatan beban kerja jaringan. Ditunjukkan bahwa peningkatan beban kerja jaringan akan meningkatkan pula *throughput* antara node sumber dan node tujuan. Pada saat 5 koneksi maksimal sumber CBR, rata-rata *throughput* antara node sumber ke node tujuan adalah 380 bps, saat 10 koneksi maksimal sumber CBR 530 bps dan 650 bps saat beban kerja simulasi maksimum yakni 15 koneksi sumber CBR. Perilaku produksi *throughput* antar node sumber ke node tujuan protokol ini mirip dengan perilaku yang sama pada protokol lain. Karena dalam hal perawatan rute yang dilakukan oleh protokol FSR juga terbantu dengan meningkatnya jumlah node yang saling berkomunikasi.

4.3. Perbandingan dari Protokol-protokol Rute

Dari hasil simulasi yang telah dilakukan dan diamati, dapat di nilai unjuk kerja masing-masing protokol berdasarkan perilaku pada saat perubahan gerakan node dalam jaringan dan saat terjadi perubahan beban kerja jaringan. Protokol rute OLSR hasil simulasi merupakan protokol yang memiliki Kualitas Rute mencapai nilai terbaik dalam penyampaian paket. Nilai tersebut diperoleh karena ia mempunyai keunggulan keberhasilan penyampaian rute, waktu tunda penyampaian paket rendah dan *throughput* yang dihasilkan. Dibandingkan dengan protokol rute FSR yang dianalisa pada simulasi.



Efisiensi protokol yang diukur berdasarkan metrik beban kerja rute untuk mendapatkan kinerja keberhasilan penyampaian paket terbaik dari protokol-protokol yang diperbandingkan. Semakin banyak dikirimkan paket kontrol rute, akan semakin banyak spasi *bandwidth* yang digunakan, karenanya beban jaringan akan semakin meningkat pula. Nilai terbaik sebesar 8 diperoleh oleh protokol OLSR, artinya saat topologi berubah cepat seiring kecepatan gerakan node pada jaringan dan perubahan beban kerja jaringan, paket kontrol rute yang dikirimkan juga sedikit.

Tabel 4.10. Tabel perbandingan routing protokol proactive

Rata – rata	OLSR	FSR
Keberhasilan menyampaikan paket	0,702	0,928
Waktu tunda node sumber – tujuan	0,603	0,046
<i>Throughput</i> node sumber – tujuan	0,68	0,52

Perbandingan unjuk kerja protokol menggunakan gabungan metrik yang mencirikan kinerja protokol sebagai berikut:

- **Kualitas Rute.** Kualitas rute ditentukan oleh *throughput*, keberhasilan menyampaikan paket dan waktu tunda penyampaian paket.
- **Akurasi Protokol.** Akurasi protokol ditentukan oleh keberhasilan penyampaian paket.
- **Efisiensi Protokol.** Efisiensi protokol ditentukan oleh bagaimana protokol menggunakan sumber daya jaringan untuk menentukan hasil kerjanya. Ditentukan oleh beban kerja rute, beban jaringan dan optimalitas jalur.
-

Tabel 4.11. Rangkuman penilaian terhadap protokol-protokol rute

<i>Perubahan Kecepatan</i>	OLSR	FSR
Kualitas rute	7,5	11
Akurasi protokol	2,5	3
Efisiensi protokol	4	3

BAB V

PENUTUP

Pada tesis ini telah dievaluasi unjuk kerja dari routing protocol proaktif yang digunakan pada jaringan *wireless ad hoc* multihop: OLSR dan FSR dengan menggunakan sebuah alat pensimulasian jaringan level-paket: *Network Simulator ns-2*. Eksperimen dilakukan menggunakan trafik laju bit konstan (CBR) digunakan untuk mengukur distribusi beban jaringan dari protokol-protokol tersebut. Masing-masing protokol disimulasikan untuk jaringan *wireless ad hoc* dengan 100 node bergerak random di dalam area geografis datar seluas 1000 x 1000 meter persegi.

5.1 Kesimpulan

Dari hasil penelitian yang dilakukan kecepatan manajemen perawatan rute merupakan faktor penting yang mempengaruhi pengukuran unjuk kerja, khususnya dalam hal laju keberhasilan penyampaian paket pada beban kerja jaringan tinggi dan peningkatan kecepatan gerakan node. Untuk membandingkan unjuk kerja routing protocol proaktif akan mendapat penilaian yang terbaik kualitas routing protocol yang diteliti berdasarkan keberhasilan penyampaian paket, *throughput* dan End-to-End delay adalah adalah: *Optimized Link State Routing* (OLSR) dan *Fisheye State Routing* (FSR)

1. *Optimized Link State Routing* (OLSR) merupakan protokol rute yang memiliki unjuk kerja akurasi protokol dengan dasar kinerja keberhasilan penyampaian paket dari node sumber ke node tujuan terbaik. Rata-rata keberhasilan penyampaian paket mencapai 99,5 %, juga merupakan protokol dengan unjuk kerja efisiensi protokol paling baik. Karena sifat protokol OLSR dalam merawat

rute dan cenderung pasif karena menggunakan dasar penentuan rute dengan tabel. Routing OLSR selalu mengupdate semua jumlah node selama waktu simulasi, dalam teknik penyampaian paket akan membentuk Multipoint Relay (MPR) dan untuk hop lebih dari 2 akan membentuk MPRS (Multipoint Relay Selector), sehingga jumlah paket terkirim sama dengan jumlah paket yang dibangkitkan.

2. *Fisheye State Routing (FSR)* merupakan protokol rute yang memiliki unjuk kerja tunda waktu penyampaian paket antara node sumber ke node tujuan terpendek. Routing protocol FSR akan mengirimkan paketnya kepada node-node yang terdekat dengan node pusat, dan kemudian baru akan mengirimkan ke node-node pada level berikutnya sesuai dengan rute pada simulasi.
3. Hasil perbandingan unjuk kerja *routing protocol wireless ad hoc* multihop, OLSR merupakan routing yang paling bagus pada rata-rata throughput dan efisiensi protokol.
4. Hasil penelitian yang telah dilakukan masih belum sempurna, akan lebih baik jika pembangkitan pola gerakan node dilakukan berulang kali untuk digunakan pada proses simulasi.

5.2 Saran

Penelitian ini pada prinsipnya masih jauh dari sempurna, semata-mata hanya karena masalah waktu dan dana yang terbatas. Untuk itu disarankan:

1. Agar ada yang melakukan penelitian lebih lanjut tentang unjuk kerja routing protocol proaktif pada jaringan *wireless ad hoc* multihop

2. Agar ada yang meningkatkan penggunaan *Network Simulator ns-2* untuk penelitian terutama yang meneliti routing pada jaringan wireless, kinerja jaringan dengan kabel, maupun komunikasi data menggunakan satelit.

DAFTAR PUSTAKA

- [1]. Shree Murthy & J.J. Garcia Luna Aceves, A Routing Protocol for Packet Radio Networks, *MOBICOM Paper ACM*, Berkeley CA, 1995.
- [2]. Lidong Zhou and Zygmunt J. Haas, Securing Ad Hoc Networks, *IEEE Network Magazine*, Vol. 13, No. 6, November/December 1999.
- [3]. Xiaoyan Hong, Kaixin Xu, and Mario Gerla, Scalable Routing Protocols for Mobile Ad Hoc Networks, University of California at Los Angeles, 2002.
- [4]. Mario Gerla, UCLA, Xiaoyan Hong, UCLA, Fisheye State Routing Protocol (FSR) for Ad Hoc Networks, *Computer Science Departement*, Rockwell Scientific Company December 17, 2001
- [5]. Thomas Clausen, Anis Laouiti, Paul Muhlethaler, Amir Qayyum, Optimized Link State Routing Protocol (OLSR), INRIA Rocquencourt, France 1 September 2001
- [6]. Fred Halsall. 1995. Data Communications, Computer Networks and Open Systems. Addison-Wesley Publishing Company
- [7]. Cisco Systems, et. Al., Internetworking Technologies Handbook 3rd Edition, Cisco Press, Indianapolis, 2000.
- [8]. Zhenping Zuo, In-Building Wireless LANs, 1999, http://www.cis.ohio-state.edu/~jain/cis788-99/wireless_lans/index.html
- [9]. Emre Çelebi, Performance Evaluation of Mobile Ad-hoc Network Routing Protocols, Bogazicy University, Istanbul, 2002 <http://cis.poly.edu/~ecelebi>
- [10]. Aaron Kershenbaum, Telecommunications Network Design Algorithms, McGraw-Hill Books Co. Singapore, 1993.
- [11]. J.J. Garcia-Lunes-Aceves, Loop-free Routing Using Diffusing Computation, *IEEE/ACM Transection on Networking*, Vol. I, No. I, February 1993
- [12]. William Stallings, High Speed Networks : TCP/IP and ATM Design Principles, Prentice-Hall Inc., New Jersey, 1998.
- [13]. Theodore S. Rappaport, Wireless Communications Principle and Practice,

Prentice-Hall PTR, New Jersey, 1996.

- [14]. Ian Glover & Peter Grant, Digital Communications, Prentice Hall Europe, 1998.
- [15]. John G. Proakis, Digital Communications, 3 rd Edition, International Edition, McGraw-Hill Inc. Singapore, 1995.
- [16]. Wenbo Fu and Dr. Stephan, Routing Mechanisms in Ad-Hoc Networks, University of Stuttgart, 2003
- [17]. Wismanu Susetiyo, Evaluasi Unjuk Kerja Protokol Rute pada Jaringan Wireless Ad Hoc Multihop, Institut Teknologi Sepuluh Nopember (ITS), Thesis 2004.
- [18]. Guangyu Pei, Mario Gerla, and Tsu-Wei Chen, Fisheye State Routing in Mobile Ad Hoc Networks, *Computer Science Departement*, University of California, Los Angeles, 2000.
- [19]. Cisco Systems, et. Al., Internetworking Technologies Handbook 3 rd Edition, Cisco Press, Indianapolis, 2000.
- [20]. Emre Çelebi, Performance Evaluation of Mobile Ad-hoc Network Routing Protocols, Bogazicy University, Istanbul, 2002 <http://cis.poly.edu/~ecelebi>
- [21]. The Network Simulator – ns-2, <http://www.isi.edu/nsnam/ns/>
- [22]. The VINT Project, The ns Manual (formely ns Notes and Documentation), May 2003, <http://www.isi.edu/nsnam/ns/ns-documentation.html>
- [23]. Theodore S. Rappaport, Wireless Communications Principle and Practice, Prentice-Hall PTR, New Jersey, 1996.
- [24]. J. Broch, D. Maltz, D. Johnson, Y-C Hu, and J. Jetcheva, A Performance Comparison of Multi-Hop Ad Hoc Network Routing Protocols, *In Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 98)*, ACM, Dallas, TX, Oct. 1998.
- [25]. William Su, Sung-Ju Lee, and Mario Gerla, Mobility Prediction in Wireless Networks, *Proceedings of IEEE MILCOM2000*, Los Angeles, California, October 2000.

LAMPIRAN I

NETWORK SIMULATOR

1. Pengaturan awal Objek Baku

Objek baku diatur pada skrip OTCL sebelum mulai membuat komponen-komponen jaringan. Ukuran-ukuran baku yang diberikan sebagian besar diperoleh dari skrip-skrip validasi *Network Simulator (ns-2)*. Berikut ini adalah pengaturan objek baku contoh bagian dari skrip otcl *ns-2* yang dipergunakan pada simulasi:

```
# =====
# Define options
# =====
set opt(chan)          Channel/WirelessChannel ;# channel type
set opt(prop)          Propagation/TwoRayGround ;# radio-propagation model
set opt(netif)         Phy/WirelessPhy ;# network interface type
set opt(mac)           Mac/802_11 ;# MAC type
set opt(ifq)           Queue/DropTail/PriQueue ;# interface queue type
set opt(ll)            LL ;# link layer type
set opt(ant)           Antenna/OmniAntenna ;# antenna model
set opt(ifqlen)        50 ;# max packet in ifq
set opt(nn)            100 ;# number of mobilenodes
set opt(adhocRouting)  OLSR ;# routing protocol

set opt(cp)            "" ;# connection pattern file
set opt(sc)            "" ;# node movement file.

set opt(x)             1000 ;# x coordinate of topology
set opt(y)             1000 ;# y coordinate of topology
set opt(seed)          0.0 ;# seed for random number gen.
set opt(stop)          500.0 ;# time to stop simulation

set opt(cbr-start)     30.0
#
=====
```



```

Mac/802_11 set CWMin_          31
Mac/802_11 set CWMax_         1023
Mac/802_11 set SlotTime_      0.000020      ;# 20us
Mac/802_11 set SIFS_          0.000010      ;# 10us
Mac/802_11 set PreambleLength_ 144          ;# 144 bit
Mac/802_11 set PLCPHeaderLength_ 48          ;# 48 bits
Mac/802_11 set PLCPDataRate_  1.0e6          ;# 1Mbps
Mac/802_11 set RTSThreshold_   0              ;# bytes
Mac/802_11 set ShortRetryLimit_ 7              ;# retransmissions
Mac/802_11 set LongRetryLimit_ 4              ;# retransmissions

LL set mindelay_              50us          ;# tunda waktu lap. Link
LL set delay_                  25us
LL set bandwidth_              0              ;# tidak digunakan
LL set off_prune_              0              ;# tidak digunakan
LL set off_CtrMcast            0              ;# tidak digunakan

```

```

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interfaces

```

```

Phy/WirelessPhy set CPTresh_ 10.0
Phy/WirelessPhy set CSTresh_ 1.559e-11
Phy/WirelessPhy set RXThresh_ 3.652e-10
Phy/WirelessPhy set bandwidth_ 2e6
Phy/WirelessPhy set Pt_ 0.28183815
Phy/WirelessPhy set freq_ 914e+6
Phy/WirelessPhy set L_ 1.0
Phy/WirelessPhy set debug_ false

```

Pengaturan ukuran paket TCP dilakukan pada bagian ini, saat simulasi digunakan sebesar 1460 byte. Berikutnya diatur macam sistem antrian yang akan digunakan simulator untuk mengatur macam antrian bagi protokol rute. Jika tidak dilakukan pengaturan, maka akan digunakan antrian prioritas *ns-2*.

```

Queue/DrotTail/PriQueue set Prefer_Routing_Protocols 1

```

Menentukan tipe antena omni-directional dengan penguatan satu:

```

# unity gain. Omni-directional antennas
# set up the antennas to be in centered the node

```

```
# and 1.5 meters above it
```

```
Antenna/OmniAntenna set X_ 0  
Antenna/OmniAntenna set Y_ 0  
Antenna/OmniAntenna set Z_ 1.5  
Antenna/OmniAntenna set Gt_ 2.0  
Antenna/OmniAntenna set Gr_ 2.0
```

Sampai saat penelitian ini dilakukan *ns-2* masih belum menggunakan aturan tiga dimensi, akan tetapi tetap valid untuk menggunakan aturan tiga dimensi untuk mengatur lokasi-lokasi node bergerak. Selanjutnya dapat diatur parameter-parameter perantara/interface emulasi untuk radio *Lucent WaveLAN* yang digunakan untuk simulasi:

```
# Initialize the SharedMedia interface with parameters to make  
# it work like the 914MHz Lucent WaveLAN DSSS radio interfaces
```

```
Phy/WirelessPhy set CPTthresh_ 10.0  
Phy/WirelessPhy set CSTthresh_ 1.559e-11  
Phy/WirelessPhy set RXThresh_ 3.652e-10  
Phy/WirelessPhy set bandwidth_ 2e6  
Phy/WirelessPhy set Pt_ 0.28183815  
Phy/WirelessPhy set freq_ 914e+6  
Phy/WirelessPhy set L_ 1.0  
Phy/WirelessPhy set debug_ false
```

Pada awal simulasi dilakukan pengaturan seting yang dilakukan untuk menginisialisasi pengaturan awal yang dilakukan untuk pengaturan awal, dan pengesetan dalam

parameter-parameter OLSR :

```
# Inisialisasi awal Simulasi  
#  
set ns_ [new Simulator]  
  
#  
# Pengesetan Parameter-2 dalam OLSR  
#  
Agent/OLSR set use_mac_ true
```

```
Agent/OLSR set debug_      false
Agent/OLSR set willingness 3
Agent/OLSR set hello_ival_ 2
Agent/OLSR set tc_ival_    5
```

Berikutnya dilakukan pengaturan dasar untuk menginisialisasi simulator pada skrip simulasi. Dan penggabungan definisi awal topografi bidang datar ke simulator

```
# Menciptakan Object Topologi dalam NAM
#
set topo [new Topography]

#
# Flat Topologi
#
$topo load_flatgrid $opt(x) $opt(y)
```

Selanjutnya simulator diperintahkan untuk melakukan penelusuran perjalanan paket untuk level kecepatan rendah dengan mode baru, serta menyimpan dalam bentuk file (tr) dan Network Animator (NAM) sebagai berikut:

```
#
set tracefd [open olsr-200.tr w]
set namtrace [open olsr-200.nam w]
$ns_ trace-all $tracefd
$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)
```

Dilakukan inisialisasi *global operation director (god)* bagi sejumlah node bergerak (nn) dan menentukan kanal radio yang digunakan.

```
set god_ [create-god $opt(nn)]
set chan_1_ [new $opt(chan)]
```

Untuk mengatur ukuran baku tipikal bagi masing-masing node bergerak diatur sebagai berikut:


```

$ns_ node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    #-channelType $opt(chan) \
    -topoInstance $topo \
    -wiredRouting OFF \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace OFF \
    -channel $chan_(1)

```

Setelah semua pengaturan awal diberikan, berikutnya dibuat ketentuan untuk node-node yang disimulasi

```

for {set i 0} {$i < $opt(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0
}

```

Mengatur agar simulasi berhenti setelah mencapai lama waktu simulasi, dengan menampilkan "NS EXITING" setelah berakhir.

```

for {set i 0} {$i < $opt(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0
}
$ns_ at $opt(stop).00000001 "puts \"NS EXITING...\" ; $ns_ halt"

```

Menguji apakah file-file skrip pola gerakan node dan pola komunikasi sudah benar dipasang, diuji dengan perintah sebagai berikut:

```

if { $opt(cp) == "" } {

```

```

    puts "*** NOTE: no connection pattern specified."
    set opt(cp) "none"
} else {
    puts "Loading connection pattern..."
    source $opt(cp)
}

if { $opt(sc) == "" } {
    puts "*** NOTE: no scenario file specified."
    set opt(sc) "none"
} else {
    puts "Loading scenario file..."
    source $opt(sc)
    puts "Load complete..."
}

```

Akhir dari skrip OTCL simulator adalah menjalankan simulator, dapat ditambahkan menuliskan beberapa nama pengaturan pada awal file penelusuran yang dibuat, dengan diberi perintah sebagai berikut:

```

puts $tracefd "M 0.0 nn $opt(nn) x $opt(x) y $opt(y) rp $opt(rp)"
puts $tracefd "M 0.0 sc $opt(sc) cp $opt(cp) seed $opt(seed)"
puts $tracefd "M 0.0 prop $opt(prop) ant $opt(ant)"

puts "Starting Simulation..."
$ns_ run

```

2. Menentukan pola trafik komunikasi dan pola gerakan node

2.1. Pola Trafik Komunikasi

Koneksi secara acak trafik TCP dan CBR antara node-node dalam jaringan dapat dibangkitkan dengan menggunakan skrip traffic-scenario generator yang ada pada ns/indep-utils/cmu-scen-gen file: cbrgen.tcl. Perintah yang diberikan pada saat membangkitkan pola trafik, sintaksisnya adalah sebagai berikut:

```

ns cbrgen.tcl |-type cbr|tcp| |-nn nodes| |-seed seed| |-mc
connection| |-rate rate| > |outdir/filename|

# Inisialisasi Agen dan Trafik Generator
#
set udp [new Agent/UDP]

```

```

set null1 [new Agent/Null]
#set null2 [new Agent/Null]
$ns_ attach-agent $node_(0) $udp
$ns_ attach-agent $node_(24) $null1
#$ns_ attach-agent $node_(23) $null2
$ns_ connect $udp $null1
#$ns_ connect $udp $null2
set cbr1 [new Application/Traffic/CBR]
#set cbr2 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 512
$cbr1 set rate_ 200Kb
$cbr1 attach-agent $udp
#$cbr2 attach-agent $udp
#$cbr2 set packetSize_ 512
#$cbr2 set rate_ 200Kb
$ns_ at $opt(cbr-start) "$cbr1 start"
#$ns_ at $opt(cbr-start) "$cbr2 start"

```

2.2. Pola Gerakan Node

Pembangkit pola gerakan node tersedia di dalam *network simulator* pada sub direktori `ns/indep-utils/cmu-scen-gen/setdest` yang berisi file-file `setdest[.cc, .h]` dan `Makefile`. Perintah untuk membangkitkan pola gerakan node dapat dilihat seperti:

```

./setdest |-n num_of_nodes| |-p pausetime| |-s maxspeed| |-t
simulat-time| |-x max_x| |-y max_y| > |outdir/movementfile|

```

Model gerakan node yang digunakan untuk menunjang simulasi, menggunakan 100 node untuk waktu jeda (pause time) 0, 40, 100, 200 dan 400 detik, kecepatan maksimum 20 meter perdetik dalam batas-batas topologi 1000 x 1000 meter untuk lama waktu simulasi 500 detik.

1. Bagian pengesetan posisi awal node:

```
#
# nodes: 100, pause:
#
# posisi tiap Node
#
$node_(0) set X_ 500.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 750.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0

$node_(2) set X_ 250.0
$node_(2) set Y_ 500.0
$node_(2) set Z_ 0.0

$node_(3) set X_ 500.0
$node_(3) set Y_ 250.0
$node_(3) set Z_ 0.0

$node_(4) set X_ 500.0
$node_(4) set Y_ 750.0
$node_(4) set Z_ 0.0

$node_(5) set X_ 676.7766952966
$node_(5) set Y_ 676.7766952966
$node_(5) set Z_ 0.0

$node_(6) set X_ 676.7766952966
$node_(6) set Y_ 323.2233047033
$node_(6) set Z_ 0.0

$node_(7) set X_ 323.2233047033
$node_(7) set Y_ 323.2233047033
$node_(7) set Z_ 0.0

$node_(8) set X_ 323.2233047033
$node_(8) set Y_ 676.7766952966
$node_(8) set Z_ 0.0

$node_(9) set X_ 1000.0
$node_(9) set Y_ 500.0
$node_(9) set Z_ 0.0

$node_(10) set X_ 691.3417162
$node_(10) set Y_ 38.0602337443
$node_(10) set Z_ 0.0

$node_(11) set X_ 853.55339059
$node_(11) set Y_ 146.44660940
$node_(11) set Z_ 0.0

$node_(12) set X_ 961.93976625
```

```
$node_(12) set Y_ 308.65828381
$node_(12) set Z_ 0.0

$node_(13) set X_ 961.93976625
$node_(13) set Y_ 691.34171618
$node_(13) set Z_ 0.0

$node_(14) set X_ 853.55339059
$node_(14) set Y_ 853.55339059
$node_(14) set Z_ 0.0

$node_(15) set X_ 691.34171618
$node_(15) set Y_ 961.93976625
$node_(15) set Z_ 0.0

$node_(16) set X_ 500.0
$node_(16) set Y_ 1000.0
$node_(16) set Z_ 0.0

$node_(17) set X_ 308.65828381
$node_(17) set Y_ 961.93976625
$node_(17) set Z_ 0.0

$node_(18) set X_ 146.44660940
$node_(18) set Y_ 853.55339059
$node_(18) set Z_ 0.0

$node_(19) set X_ 38.0602337443
$node_(19) set Y_ 691.34171618
$node_(19) set Z_ 0.0

$node_(20) set X_ 0.0
$node_(20) set Y_ 500.0
$node_(20) set Z_ 0.0

$node_(21) set X_ 38.0602337443
$node_(21) set Y_ 308.65828381
$node_(21) set Z_ 0.0

$node_(22) set X_ 146.44660940
$node_(22) set Y_ 146.44660940
$node_(22) set Z_ 0.0

$node_(23) set X_ 308.65828381
$node_(23) set Y_ 38.0602337443
$node_(23) set Z_ 0.0

$node_(24) set X_ 500.0
$node_(24) set Y_ 0.0
$node_(24) set Z_ 0.0

$node_(25) set X_ 0.0
$node_(25) set Y_ 0.0
$node_(25) set Z_ 0.0
.
.
.
dan seterusnya sampai 100 node
```

2. Bagian memulai melakukan gerakan awal

```
# Pergerakan Node
#
$ns_ at 0.000000000000 "$node_(24) setdest 482.463841874264
352.861616526548 16.418856916745"
$ns_ at 0.000000000000 "$node_(23) setdest 235.593109625244
376.846774617240 13.734250258823"
$ns_ at 0.000000000000 "$node_(22) setdest 361.170229297925
381.465956881856 14.411052371724"
$ns_ at 0.000000000000 "$node_(21) setdest 866.260429886675
90.791742849323 14.744153834596"
$ns_ at 0.000000000000 "$node_(20) setdest 50.518292353443
163.867023513732 15.320105936048"
$ns_ at 0.000000000000 "$node_(19) setdest 577.054657281230
552.101984302796 13.611582119765"
$ns_ at 0.000000000000 "$node_(18) setdest 327.272437995455
16.847490683263 10.019595107081"
$ns_ at 0.000000000000 "$node_(17) setdest 69.759765758652
465.692521367232 5.546016654366"
$ns_ at 0.000000000000 "$node_(16) setdest 803.993457516339
99.698916435562 0.355375603288"
$ns_ at 0.000000000000 "$node_(15) setdest 777.096476786786
532.647003976297 12.600501920307"
$ns_ at 0.000000000000 "$node_(14) setdest 818.383058585151
338.842585375360 18.856238020141"
$ns_ at 0.000000000000 "$node_(13) setdest 589.521238026306
474.688759663902 19.801461286541"
$ns_ at 0.000000000000 "$node_(12) setdest 68.621714663067
376.267029499529 14.182945324586"
$ns_ at 0.000000000000 "$node_(11) setdest 582.463841874264
352.861616526548 6.418856916745"
$ns_ at 0.000000000000 "$node_(10) setdest 335.593109625244
36.846774617240 13.734250258823"
$ns_ at 0.000000000000 "$node_(9) setdest 361.170229297925
581.465956881856 14.411052371724"
$ns_ at 0.000000000000 "$node_(8) setdest 66.260429886675
910.791742849323 14.744153834596"
$ns_ at 0.000000000000 "$node_(7) setdest 750.518292353443
163.867023513732 15.320105936048"
$ns_ at 0.000000000000 "$node_(6) setdest 57.054657281230
552.101984302796 13.611582119765"
$ns_ at 0.000000000000 "$node_(5) setdest 37.272437995455
616.847490683263 10.019595107081"
$ns_ at 0.000000000000 "$node_(4) setdest 669.759765758652
485.692521367232 5.546016654366"
$ns_ at 0.000000000000 "$node_(3) setdest 703.993457516339
990.698916435562 0.355375603288"
$ns_ at 0.000000000000 "$node_(2) setdest 577.096476786786
552.647003976297 12.600501920307"
$ns_ at 0.000000000000 "$node_(1) setdest 88.383058585151
738.842585375360 18.856238020141"
$ns_ at 0.000000000000 "$node_(0) setdest 199.521238026306
199.688759663902 19.801461286541".
.
.
dan seterusnya sampai node 99
```


3. *Bagian seting global operation director (god)*

```
$god_ set-dist 0 0 3
$god_ set-dist 0 1 3
$god_ set-dist 0 2 4
$god_ set-dist 0 3 4
$god_ set-dist 0 4 1
$god_ set-dist 0 5 4
$god_ set-dist 0 6 4
$god_ set-dist 0 7 1
$god_ set-dist 0 8 3
$god_ set-dist 0 9 3
$god_ set-dist 0 10 3
$god_ set-dist 0 11 4
$god_ set-dist 0 12 1
$god_ set-dist 0 13 4
$god_ set-dist 0 14 1
$god_ set-dist 0 15 1
$god_ set-dist 0 16 3
$god_ set-dist 0 17 2
$god_ set-dist 0 18 3
$god_ set-dist 0 19 1
$god_ set-dist 0 20 2
$god_ set-dist 0 21 1
$god_ set-dist 0 22 2
$god_ set-dist 0 23 3
.
.
.
```

dan seterusnya sampai node ke-100

4. *Menentukan pergerakan node*

```
#
if { $opt(cp) == "" } {
    puts "**** NOTE: no connection pattern specified."
    set opt(cp) "none"
} else {
    puts "Loading connection pattern..."
    source $opt(cp)
}
if { $opt(sc) == "" } {
    puts "**** NOTE: no scenario file specified."
    set opt(sc) "none"
} else {
    puts "Loading scenario file..."
    source $opt(sc)
    puts "Load complete..."
}
.
.
.
```

dan seterusnya sampai waktu simulasi berakhir....

Dalam simulasi pergerakan node lebih awal akan menentukan posisi dan inisialisasi pada tiap node dalam pembentukan NAM adalah :

```
#
for {set i 0} {$i < $opt(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 20
}
```

Dari proses pembangkitan pola gerakan node random menggunakan utilitas yang sudah tersedia pada *ns-2* akan terdapat pola yang berbeda setiap dilakukan pembangkitan. Untuk memperhalus hasil simulasi maka diperlukan pembangkitan lebih dari satu pola gerakan untuk waktu jeda yang sama. Perubahan koneksi yang bisa dilihat dari pengaturan global operation director juga dihitung oleh program pembangkit pola trafik, dan akan memberi tahukan terhadap semua node apabila simulasi telah berakhir :

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

$ns_ at $opt(stop).0002 "puts \"NS EXITING...\" ; $ns_ halt"
$ns_ at $opt(stop).0001 "stop"

proc stop {} {
    global ns_ tracefd namtrace
    $ns_ flush-trace
    close $tracefd
    close $namtrace
}
```

LAMPIRAN II

TABULASI DATA PENELITIAN KOMPARASI WIRELESS AD HOC

ROUTING PROTOCOLS

Koneksi Maksimum = 26; Jumlah Node = 100; Waktu Simulasi = 500

Protokol Rute		OLSR (Optimized Link State Routing)				
No.	Waktu jeda (detik)	0	40	100	200	400
Pengukuran & Perhitungan						
1	Jumlah node pengirim	26	26	26	26	26
2	Jumlah node penerima	26	26	26	26	26
3	Jumlah paket yang dibangkitkan	23519	23519	23519	23519	23519
4	Jumlah paket yang dikirimkan	23519	23519	23519	23518	23519
5	Jumlah paket yang diforward	0	0	0	0	0
6	Jumlah paket yang didrop	1	1	1	1	1
7	Jumlah paket yang lost	0	0	0	0	0
8	Ukuran paket minimal	512	512	512	512	512
9	Ukuran paket maksimal	552	552	552	552	552
10	Ukuran paket rata-rata	465.8424	465.8424	465.8424	465.8424	465.8424
11	Data terkirim (bytes)	12371556	12371556	12371556	12371556	12371556
12	Data diforward (bytes)	0	0	0	0	0
13	Data didrop (bytes)	1	1	1	1	1
14	Waktu Tunda Rata-rata End2end (detik)	0.00586353	0.00586353	0.00586353	0.00586353	0.00586353

Protokol Rute		FSR (Fisheye State Routing)				
No.	Waktu jeda (detik)	0	40	100	200	400
Pengukuran & Perhitungan						
1	Jumlah node pengirim	26	26	26	26	26
2	Jumlah node penerima	26	26	26	26	26
3	Jumlah paket yang dibangkitkan	17968	17968	17967	17968	17968
4	Jumlah paket yang dikirimkan	17956	17947	17848	17955	17850
6	Jumlah paket yang didrop	12	21	119	13	118
8	Ukuran paket minimal	24	24	24	24	24
9	Ukuran paket maksimal	1256	1308	1244	1328	1344
10	Ukuran paket rata-rata	497.912	497.8946	497.8717	498.5087	497.9927
11	Data terkirim (bytes)	9504544	9499760	9447404	9504728	9448392
13	Data didrop (bytes)	6383	11172	63308	6916	62776