



UNIVERSITAS INDONESIA

**ANALISIS *DELAY HANDOVER* PADA JARINGAN INTERNET DI  
KERETA API BERDASARKAN PROTOKOL *MOBILE IP***

**SKRIPSI**

**WAHYU ADIGUNA**  
**04 05 03 0796**

**FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
DESEMBER 2009**



**UNIVERSITAS INDONESIA**

**ANALISIS *DELAY HANDOVER* PADA JARINGAN INTERNET DI  
KERETA API BERDASARKAN PROTOKOL *MOBILE IP***

**SKRIPSI**

**Diajukan untuk melengkapi  
sebagian persyaratan menjadi Sarjana Teknik**

**WAHYU ADIGUNA  
04 05 03 0796**

**FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
DESEMBER 2009**

## HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun yang dirujuk  
telah saya nyatakan dengan benar.

Nama : Wahyu Adiguna

NPM : 0405030796

Tanda Tangan : 

Tanggal : 15 Desember 2009

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Wahyu Adiguna  
NPM : 0405030796  
Program Studi : Teknik Elektro  
Judul Skripsi : Analisis *Delay Handover* pada Jaringan Internet di Kereta  
Api Berdasarkan Protokol *Mobile IP*

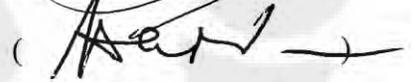
**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.**

## DEWAN PENGUJI

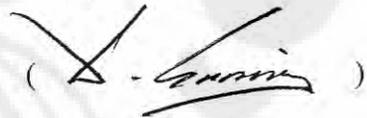
Pembimbing : Ir. Muhamad Asvial, M.sc, Ph.D



Penguji : Prof. Dr. Ir. Bagio Budiardjo, M.sc



Penguji : Prof. Dr. Ir. Dadang Gunawan, M.Eng



Ditetapkan di : Depok

Tanggal : 30 Desember 2009

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Ir. Muhamad Asvial, M.sc, Ph.D, selaku dosen pembimbing yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan, bimbingan, dan diskusi sehingga skripsi ini dapat terselesaikan dengan baik;
2. Orang tua dan keluarga tercinta yang telah memberikan bantuan doa, moril, maupun materiil;
3. Sahabat-sahabat dekat atas dukungan moril yang telah diberikan;
4. Rekan-rekan di Departemen Teknik Elektro angkatan 2005 yang telah banyak membantu dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 15 Desember 2009

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR  
UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Wahyu Adiguna  
NPM : 0405030796  
Program Studi : Teknik Elektro  
Fakultas : Teknik  
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul:

***Analisis Delay Handover* pada Jaringan Internet di Kereta Api Berdasarkan  
Protokol Mobile IP**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada tanggal : 15 Desember 2009  
Yang menyatakan



(Wahyu Adiguna)

## ABSTRAK

Nama : Wahyu Adiguna  
Program Studi : Teknik Elektro  
Judul : Analisis *Delay Handover* pada Jaringan Internet di Kereta Api Berdasarkan Protokol *Mobile IP*

Internet di kereta api merupakan suatu konsep yang sedang marak dalam beberapa tahun terakhir. Percobaan di beberapa Negara telah membuktikan kemungkinan adanya akses internet bagi penumpang kereta api, namun tak satu pun dari percobaan tersebut yang menggabungkan akses *broadband*, skalabilitas, *handover* yang mulus, dan jaminan kualitas layanan dalam satu solusi. Oleh karena itu, dibutuhkan suatu arsitektur jaringan internet baru dan solusi mobilitas antarkerja yang memungkinkan. Salah satu mobilitas antarkerja yang memungkinkan adalah *Mobile IP* (MIP). Pada skripsi ini, akan disimulasikan bagaimana MIP menangani *handover* dan menjamin paket sampai ke tujuan. Hasil simulasi menunjukkan paket dapat terkirim.

Kata kunci:

*Foreign Agent* (FA), *Handover*, *Home Agent* (HA), *Mobile IP* (MIP), *Mobile Host* (MH)

## ABSTRACT

Name : Wahyu Adiguna  
Study Program : Electrical Engineering  
Title : Handover Delay Analysis of Internet Network on Train Based on Mobile IP Protocol

Internet-on-the-train is a rising concept in the last few years. Several trials in different countries have proved the feasibility of offering internet access to train commuters, but none of them combines broadband access, scalability, seamless handover and quality of service guarantees in one solution. Therefore, new internet network architecture and possible inter-working mobility solution is needed. One of possible inter-working mobility solution is Mobile IP (MIP). In this thesis, how MIP handle handover and ensure packets to reach its destination will be simulated. The result shows that packets are successfully delivered.

Key words:

Foreign Agent (FA), Handover, Home Agent (HA), Mobile IP (MIP), Mobile Host (MH)

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN ORISINALITAS.....	ii
HALAMAN PENGESAHAN.....	iii
KATA PENGANTAR.....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR.....	v
ABSTRAK.....	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xi
DAFTAR ISTILAH.....	xii
<b>BAB 1 PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Tujuan Penelitian.....	2
1.3 Rumusan dan Batasan Masalah.....	3
1.4 Metodologi Penelitian dan Sistematika Penulisan.....	3
<b>BAB 2 QoS: ARSITEKTUR JARINGAN INTERNET DI KERETA API</b> .....	<b>5</b>
2.1 Kebutuhan <i>Bandwidth</i> .....	5
2.2 <i>Handover</i> .....	6
2.2.1 Proses <i>Handover</i> .....	6
2.2.2 Tujuan <i>Handover</i> .....	7
2.2.3 Prioritas <i>Handover</i> .....	8
2.2.4 Tipe <i>Handover</i> .....	8
2.2.5 Permasalahan <i>Handover</i> .....	9
2.3 Jaringan Komunikasi <i>Mobile</i> .....	9
2.3.1 <i>General Packet Radio Service</i> (GPRS).....	9
2.3.2 <i>Enhanced Data rates for GSM Evolution</i> (EDGE).....	11
2.3.3 <i>Universal Mobile Telecommunication System</i> (UMTS).....	12
2.3.4 <i>High-Speed Downlink Packet Access</i> (HSDPA).....	13
2.4 Kualitas Layanan (QoS).....	13
2.4.1 Performansi.....	14
2.4.2 Diferensiasi.....	15
2.4.3 QoS dalam IP.....	15
2.5 Performansi Jaringan Komunikasi <i>Mobile</i> .....	16
2.6 Arsitektur.....	17
2.6.1 <i>Policy Decision Function</i> .....	18
2.6.2 <i>Mobility Management</i> .....	18
2.7 Tinjauan <i>Mobile IP</i> (MIP).....	19
2.7.1 Prinsip Kerja <i>Mobile IP</i> (MIP).....	20
2.7.2 <i>Overhead</i> Pada <i>Mobile IP</i> (MIP).....	22
2.8 Efek <i>Doppler</i> .....	23

Universitas Indonesia

<b>BAB 3 PERANCANGAN MODEL SIMULASI .....</b>	<b>24</b>
3.1 Gambaran Umum Simulasi .....	24
3.2 Network Simulator .....	24
3.2.1 Event Scheduler .....	25
3.2.2 Komponen Jaringan .....	26
3.2.2.1 <i>Node</i> .....	27
3.2.2.2 <i>Link</i> .....	27
3.2.2.3 <i>Antrean</i> .....	28
3.2.2.4 <i>Agent</i> .....	28
3.2.2.5 <i>Aplikasi</i> .....	28
3.2.2.6 <i>Tracing</i> .....	28
3.3 Perancangan Simulasi .....	29
3.4 Alur Simulasi .....	30
3.5 Parameter Simulasi .....	32
<b>BAB 4 ANALISIS SIMULASI .....</b>	<b>33</b>
4.1 Proses Simulasi .....	33
4.2 Pengamatan Paket .....	37
4.2.1 <i>Graph Bandwidth</i> .....	37
4.2.1.1 <i>Bandwidth Node 1 – Node 2</i> .....	37
4.2.1.2 <i>Bandwidth Node 1 – Node 3</i> .....	40
4.2.1.3 <i>Bandwidth Node 2 – Node 3</i> .....	41
4.3 Variasi Kecepatan <i>Mobile Host</i> (MH) .....	42
4.3.1 <i>Bandwidth Node 1 – Node 2</i> .....	42
4.3.2 <i>Bandwidth Node 1 – Node 3</i> .....	45
4.3.3 <i>Bandwidth Node 2 – Node 3</i> .....	46
<b>BAB 5 KESIMPULAN .....</b>	<b>47</b>
DAFTAR ACUAN .....	48
DAFTAR PUSTAKA .....	49
LAMPIRAN .....	50

## DAFTAR GAMBAR

Gambar 2.1	Proses <i>handover</i> .....	7
Gambar 2.2	Arsitektur jaringan UMTS.....	13
Gambar 2.3	Arsitektur: <i>gateway</i> pada <i>central</i> dan kereta api .....	17
Gambar 2.4	MIP <i>encapsulation</i> .....	23
Gambar 3.1	Arsitektur dasar Network Simulator.....	25
Gambar 3.2	Hirarki komponen jaringan pada Network Simulator .....	26
Gambar 3.3	<i>Unicast</i> dan <i>Multicast node</i> .....	27
Gambar 3.4	Topologi jaringan <i>wired-cum-wireless</i> .....	29
Gambar 3.5	Diagram alir simulasi .....	31
Gambar 4.1	Pesan peringatan pada awal simulasi .....	33
Gambar 4.2	<i>Window</i> simulasi.....	34
Gambar 4.3	MH berpindah dari HA ke FA.....	35
Gambar 4.4	MH tiba di FA .....	35
Gambar 4.5	MH kembali ke HA .....	36
Gambar 4.6	Akhir simulasi .....	36
Gambar 4.7	Grafik <i>bandwidth</i> antara <i>node 1</i> dan <i>node 2</i> .....	37
Gambar 4.8	MH mulai meninggalkan HA .....	38
Gambar 4.9	MH tiba di FA .....	39
Gambar 4.10	MH meninggalkan FA.....	39
Gambar 4.11	MH tiba kembali di HA.....	40
Gambar 4.12	Grafik <i>bandwidth</i> antara <i>node 1</i> dan <i>node 3</i> .....	40
Gambar 4.13	Grafik <i>bandwidth</i> antara <i>node 2</i> dan <i>node 3</i> .....	41
Gambar 4.14	Grafik <i>bandwidth</i> antara <i>node 1</i> dan <i>node 2</i> .....	42
Gambar 4.15	MH mulai meninggalkan HA .....	43
Gambar 4.16	MH tiba di FA .....	43
Gambar 4.17	MH meninggalkan FA.....	44
Gambar 4.18	MH tiba kembali di HA.....	44
Gambar 4.19	Grafik <i>bandwidth</i> antara <i>node 1</i> dan <i>node 3</i> .....	45
Gambar 4.20	Grafik <i>bandwidth</i> antara <i>node 2</i> dan <i>node 3</i> .....	46

**DAFTAR TABEL**

Tabel 2.1	Kepekaan performansi untuk jenis layanan .....	15
Tabel 2.2	Perbandingan hasil pengujian GPRS, EDGE, UMTS, dan HSDPA.....	16

**Universitas Indonesia**

## DAFTAR ISTILAH

<i>Backhaul</i>	Saluran utama (tulang punggung) penghubung antarjaringan pada suatu daerah/area yang jaraknya saling berjauhan.
<i>Broadband</i>	Frekuensi jalur lebar yang mampu memindahkan lebih banyak data dan lebih tinggi dibandingkan dengan frekuensi yang lebih sempit ( <i>narrowband</i> ).
<i>Downlink</i>	Saluran untuk mengirim data keluar (ke suatu <i>server</i> di internet).
<i>Encapsulation</i>	Proses pembungkusan data pada <i>header</i> protokol tertentu.
<i>Gateway</i>	Sebuah mekanisme yang menyediakan akses ke sebuah sistem lain yang terhubung dalam sebuah jaringan.
<i>Foreign Agent</i>	<i>Router</i> pada <i>foreign network</i> . <i>Foreign agent</i> memberikan CoA yang digunakan oleh MIP.
<i>Foreign Network</i>	Jaringan yang sedang dikunjungi oleh <i>mobile node</i> ketika <i>mobile node</i> sedang berpindah.
<i>Home Agent</i>	<i>Router</i> pada <i>home network</i> yang bertanggung jawab menangkap <i>datagram</i> yang ditujukan untuk <i>mobile node</i> dan meneruskannya ketika <i>mobile node</i> berada di <i>foreign network</i> .
<i>Home Network</i>	Jaringan yang diidentifikasi oleh IP <i>address</i> permanen dari <i>mobile node</i> .
<i>Mobile Node</i>	Suatu <i>node</i> yang dapat mengganti <i>point of attachment</i> -nya ke internet. Misalnya laptop, PDA, dan sebagainya.
<i>Overhead</i>	Data ekstra yang tidak memberikan kontribusi terhadap informasi di dalam isi pesan yang muncul ketika suatu aplikasi mengirimkan data melalui suatu jaringan. Semakin kecil persentase <i>overhead</i> , maka semakin baik.

## **BAB I PENDAHULUAN**

### **1.1 Latar Belakang**

Dewasa ini, kebutuhan akan akses informasi semakin meningkat seiring dengan berkembangnya teknologi informasi dan komunikasi. Perkembangan tersebut telah memudahkan pengguna dalam memperoleh informasi dengan mudah dan cepat, terutama dengan ditemukannya teknologi nirkabel. Pengguna dapat mengakses informasi melalui internet dan dapat melakukan komunikasi dengan pengguna lainnya selama pengguna tersebut masih terkoneksi ke sistem jaringan nirkabel, kapan saja dan di mana saja, tak terkecuali di kereta api.

Beberapa teknologi nirkabel dapat digunakan untuk memungkinkan akses internet *broadband* di kereta api. Jika kita melihat kepada GPRS (dianggap sebagai layanan data *mobile* selular yang pertama), banyak Negara yang bahkan tidak memiliki cakupan nasional yang penuh. Saat mempertimbangkan UMTS atau WiMAX *cells*, saat ini mereka hanya terdapat di daerah kota-kota besar. Di sisi lain, satelit dapat menjangkau di hampir setiap lokasi geografis dikarenakan *footprint*-nya yang besar (cakupan daerah yang luas). Namun, satelit tersebut membutuhkan kondisi *line-of-sight* untuk menjaga koneksi *broadband*. Hal ini menyebabkan koneksi satelit di daerah perkotaan yang padat, terowongan, dan daerah pegunungan menjadi sulit atau tidak memungkinkan.

Temuan ini memperkuat kenyataan bahwa jaringan rel kereta api dari perusahaan yang mengoperasikannya jarang yang benar-benar sudah tercakupi sepenuhnya oleh teknologi *backhaul* nirkabel (*broadband*). Oleh karena itu, ketika kita mempertimbangkan koneksi internet di kereta api, kita perlu menggabungkan berbagai teknologi. Hal ini melibatkan pergantian dari teknologi yang satu ke yang lain saat meninggalkan daerah cakupan dari teknologi yang sebelumnya. Hal ini disebut *inter-technology handover* atau "*vertical handover*".

Membundel beberapa teknologi nirkabel *broadband* menjadi satu “pipa” data harus dilakukan secara transparan dengan cara: penumpang di kereta api tidak boleh menyadari adanya *handover* dan harus memperoleh jaringan multimedia yang mulus tanpa gangguan. Arsitektur jaringan IP yang dibahas pada skripsi ini dapat menggunakan beberapa teknologi akses nirkabel sehingga konektivitas internet yang lancar dapat tersedia di kereta api. Sedangkan untuk menangani *handover* dan permasalahan IP *address* pada *end device*, protokol mobilitas antarkerja yang memungkinkan layanan internet di kereta api, yaitu *Mobile IP* (MIP), akan diimplementasikan.

Pada skripsi ini akan disimulasikan bagaimana MIP menangani *handover* ketika kereta api, yang dianggap sebagai *mobile node*, berpindah dari satu *base station* ke *base station* yang lain. Simulasi dibuat dengan merancang suatu topologi jaringan *wired-cum-wireless* dengan menggunakan *software* Network Simulator 2.29. Kemudian, paket yang ditujukan untuk *mobile node* akan diamati dengan menggunakan fitur grafik yang terdapat pada Network Animator (NAM). Hasil simulasi menunjukkan paket dapat terkirim dengan rata-rata *delay* sebesar 34 detik pada kecepatan 20 m/s dan 20 detik pada kecepatan 33 m/s.

Dengan adanya internet di kereta api ini, diharapkan akan memberikan banyak manfaat, terutama pemanfaatan waktu, bagi awak kereta api dan penumpang yang sebagian besar aktivitasnya banyak menghabiskan waktu di kereta api.

## 1.2 Tujuan Penelitian

Tujuan yang ingin dicapai dalam skripsi ini adalah:

1. Merancang dan membuat simulasi implementasi MIP pada *mobile node*.
2. Mengetahui karakteristik, prinsip kerja, dan fitur pada MIP.
3. Mengetahui performansi MIP dalam menangani *handover*.
4. Menganalisis data, yang berupa grafik, yang diperoleh melalui simulasi.
5. Membuat kesimpulan berdasarkan analisis yang telah dilakukan.

### 1.3 Rumusan dan Batasan Masalah

Masalah yang akan diteliti pada skripsi ini berdasarkan latar belakang di atas adalah:

1. Membahas arsitektur jaringan internet di kereta api serta komponen pada *gateway* dan *Central Management System*.
2. Membahas karakteristik, prinsip kerja, dan fitur pada MIP.
3. Implementasi MIP pada *mobile node* melalui simulasi.
4. Menganalisis karakteristik dan performansi MIP dalam menangani *handover* berdasarkan hasil simulasi yang diperoleh.

Pada skripsi ini, rumusan masalah di atas dibatasi dengan ruang lingkup sebagai berikut:

1. Jenis *traffic* yang digunakan pada simulasi adalah *Constant Bit Rate* (CBR).
2. Jumlah *base station* yang digunakan dua buah.
3. Jumlah *mobile node* yang digunakan satu buah.
4. Kecepatan *mobile node* dibuat konstan sebesar 20 m/s, dimana kecepatan tersebut termasuk dalam *range* kecepatan rata-rata kereta api di Indonesia yang berkisar pada 60-110 km/jam. Selain itu, dibuat variasi kecepatan sebesar 33 m/s sebagai pembanding.

### 1.4 Metodologi Penelitian dan Sistematika Penulisan

Metodologi penelitian yang digunakan dalam skripsi ini adalah sebagai berikut:

1. Studi literatur dengan mengumpulkan bahan-bahan referensi dan mempelajari literatur-literatur yang menunjang proses penelitian, seperti jurnal-jurnal, artikel-artikel, dan *paper* mengenai arsitektur jaringan internet di kereta api, MIP, Network Simulator, dan Network Animator.
2. Analisis permasalahan dan kebutuhan yang diperlukan untuk membuat simulasi implementasi MIP.

3. Membuat simulasi implementasi MIP dengan menggunakan *software* Network Simulator 2.29.
4. Melakukan analisis terhadap hasil yang diperoleh melalui simulasi.
5. Menyusun buku laporan skripsi dan membuat kesimpulan akhir.

Sistematika penulisan skripsi ini dibagi menjadi lima bab dengan rincian sebagai berikut:

#### **BAB I      PENDAHULUAN**

Pada bab ini akan dibahas mengenai latar belakang, tujuan penelitian, ruang lingkup permasalahan, metodologi penelitian, dan sistematika penulisan skripsi.

#### **BAB II     QoS: ARSITEKTUR JARINGAN INTERNET DI KERETA API**

Pada bab ini akan dijelaskan mengenai arsitektur jaringan internet di kereta api, sistem komunikasi *mobile* pada kereta api, kebutuhan *bandwidth*, *handover*, QoS, dan teori-teori yang mendukung dan mendasari penulisan skripsi ini.

#### **BAB III    PERANCANGAN MODEL SIMULASI**

Pada bab ini akan dijelaskan mengenai perancangan model dan alur simulasi yang akan dilakukan untuk simulasi implementasi MIP pada *mobile node*.

#### **BAB IV    ANALISIS SIMULASI**

Pada bab ini akan dijelaskan mengenai proses dari jalannya simulasi serta dilakukan analisis terhadap hasil yang diperoleh melalui simulasi.

#### **BAB V     KESIMPULAN**

Pada bab ini berisikan kesimpulan mengenai permasalahan yang dibahas berdasarkan serangkaian penelitian yang dilakukan.

## **BAB II**

### **QoS: ARSITEKTUR JARINGAN INTERNET DI KERETA API**

Dalam menyediakan layanan internet di kereta api, ada beberapa hal mendasar yang harus diperhatikan. Hal tersebut antara lain:

1. Kebutuhan *bandwidth*.
2. *Handover*.
3. Jaringan komunikasi *mobile* yang digunakan sebagai *backhaul*.
4. Kualitas layanan (QoS).

Karena kereta api bergerak dengan cepat, *handover* tentu menjadi sesuatu yang sangat krusial sehingga jaringan komunikasi *mobile* yang digunakan sebagai *backhaul* antara kereta api dengan tanah harus mampu menangani *handover* yang cepat. Selain itu, jaringan komunikasi *mobile* tersebut juga harus dapat memberikan konektivitas internet yang lancar bagi penumpang dengan QoS yang diperlukan dengan prioritas *bandwidth* bagi awak kereta api.

#### **2.1 Kebutuhan *Bandwidth***

Awak kereta api tentu mendapat prioritas *bandwidth* yang lebih tinggi dibandingkan dengan penumpang. Layanan yang digunakan oleh awak kereta api antara lain *Closed-circuit Television* (CCTV) untuk memantau kondisi gerbong, komunikasi antarawak, dan *Machine-to-Machine* (M2M).

Sedangkan layanan yang dapat digunakan oleh penumpang adalah sebagai berikut:

1. Meningkatkan cakupan telepon selular.
2. *Best effort*, seperti *e-mail*, *surfing*, *Virtual Private Network* (VPN), dan *chatting*.
3. *Streaming* musik dan video.
4. *Interactivity*, seperti *Voice over IP* (VoIP), *video phony*, dan *gaming*.

## 2.2 Handover

*Handover*, *handoff*, atau *roaming* adalah proses pengalihan kanal *traffic* secara otomatis pada *Mobile Station* (MS) yang sedang digunakan untuk berkomunikasi tanpa terjadinya pemutusan hubungan. Hal ini menunjukkan bahwa *handover* pada dasarnya adalah sebuah *call* koneksi yang bergerak dari satu sel ke sel yang lain [1].

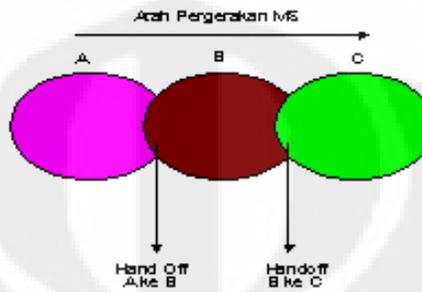
*Handover* dipengaruhi oleh faktor level daya sinyal terima (*RxLevel*), kualitas sinyal terima (*RxQual*), *power budget neighbouring cell*, dan jarak antara MS dan *Base Transceiver Station* (BTS) yang masing-masing mempunyai nilai ambang batas sehingga ketika nilai ambang batas tersebut sudah terlewati, *handover* harus dilakukan untuk menjaga suatu panggilan agar tidak terputus. Keputusan untuk melakukan *handover* tersebut dibuat oleh *Base Station Controller* (BSC), yaitu dengan mengevaluasi secara permanen pengukuran yang diambil oleh MS dan BTS. Selain itu, *handover* juga dapat terjadi apabila *traffic* dari sel yang dituju sudah penuh. Saat MS melewati sel, maka MS akan dialihkan ke *neighbouring cell* dengan beban *traffic* yang lebih kecil.

### 2.2.1 Proses Handover

Ketika MS bergerak menjauhi suatu sel, maka daya yang diterima oleh MS akan berkurang. Jika MS bergerak menjauhi *base station* (sel), maka daya pancar akan semakin berkurang. Menjauhnya MS dari sel asal akan membuat MS mendekati sel yang lain. Sel yang lain tersebut dinamakan sel kandidat, yaitu sel yang akan menerima pelimpahan MS dari sel sebelumnya.

*Mobile Station Controller* (MSC) melalui sel kandidat akan memonitor pergerakan MS dan menangkap daya pancar MS. Di antara sel kandidat yang menerima daya pancar MS terbesar, maka pelimpahan MS akan berada pada sel tersebut. Sel kandidat yang menerima pelimpahan MS akan melakukan *monitoring*. Proses *monitoring* dilakukan oleh MSC dan menginstruksikan pada sel kandidat tersebut.

Pada saat *handoff*, supervisi dipersingkat. MSC melakukan prioritas pendudukan kanal pada MS yang akan mengalami *handoff* dan sel kandidat dibuat urutan prioritas. Berikut ini adalah gambar mengenai proses terjadinya *handover*.



Gambar 2.1 Proses *handover* [1]

Proses *handover* tidak selalu berjalan lancar, walaupun nilai ambang batas sudah dilewati namun *handover* tetap tidak mau dilakukan. Hal tersebut dikarenakan beberapa faktor sehingga menyebabkan kegagalan *handover* (*failure*). Kegagalan *handover* belum tentu menyebabkan suatu panggilan terputus, bisa juga mengakibatkan kualitas suara yang diterima menjadi buruk. Panggilan terputus merupakan akibat yang paling buruk jika *handover* tidak dapat dilakukan.

### 2.2.2 Tujuan *Handover*

Secara umum, *handover* memiliki tujuan sebagai berikut:

1. *As imperceptible to user as possible*. Sedapat mungkin tidak dirasakan oleh pengguna dengan cara meminimalkan waktu *handoff* dengan menggunakan teknik interpolasi suara.
2. *As successfully as possible*. Dengan meminimalkan *error* pada saat estimasi kebutuhan *handoff*.
3. *As infrequently as possible*. MSC melakukan *assign* (*sharing*) pada kanal yang sama pada *neighbouring cell* dan meminjam kanal lain dari *neighbouring cell* pada sel sebelumnya (*MSC assigns same channel in the second cell and 'rents' another channel from the second to the first cell*).

Universitas Indonesia

### 2.2.3 Prioritas Handover

Secara umum, *handover* mempunyai prioritas sebagai berikut:

1. MSC melakukan pencarian kanal baru bagi MS yang akan melakukan *handoff* dan *internal call*.
2. Langkah terbaik adalah melakukan *blocking* MS yang baru akan aktif daripada MS yang sedang aktif.

### 2.2.4 Tipe Handover

Berikut ini adalah beberapa tipe dari *handover* [1]:

1. *Intra Cell Handover*, pemindahan informasi yang dikirim dari satu kanal ke kanal yang lain pada sel yang sama. Dilakukan karena terjadi gangguan interferensi atau operasi pemeliharaan.
2. *Intra-BSC Handover*, yaitu *handover* yang dikontrol oleh BSC. BTS yang lama dan baru sama-sama di bawah kendali sebuah BSC. *Handover* ditangani seluruhnya oleh BSC. MSC menerima informasi lokasi sel baru yang digunakan MS dari BSC.
3. *Intra-MS Handover*, yaitu *handover* yang terjadi dalam sebuah MSC. BTS lama yang baru berada di bawah sebuah MSC tapi dikendalikan oleh BSC yang berbeda.
4. *Inter-MS Handover*, yaitu *handover* antardua MSC. BTS lama dan yang baru berada pada MSC area yang berbeda.
5. *Soft Handoff*, melibatkan *inter-cell handoff* dan termasuk tipe *make before-break connection*. Koneksi antara MS dan *cell site* dilakukan oleh beberapa *cell site* selama proses *handoff*. *Soft handoff* hanya terjadi jika sel asal dan sel tujuan beroperasi pada kanal frekuensi yang sama.
6. *Softer Handoff*, adalah *intra-cell handoff* yang terjadi antarsektor dalam suatu *cell site* dan termasuk tipe koneksi *make-before-break*.

7. *Hard Handoff*, memungkinkan MS untuk berpindah dari CDMA ke sistem lainnya dan termasuk tipe koneksi *break-before-make*. *Hard handoff* juga bisa terjadi untuk dua sel CDMA yang beroperasi pada frekuensi yang berbeda.

### 2.2.5 Permasalahan *Handover*

Pada saat MS bergerak dari satu sel ke sel lainnya, *traffic* pada sel sebelumnya harus diubah ke kanal dengan *traffic* dan kanal kontrol sel yang baru. Apabila terjadi kegagalan *handover* akan berakibat *dropcall*, yaitu terputusnya hubungan saat percakapan sedang berlangsung. Faktor-faktor penyebab gagalnya *handover* antara lain:

1. Interferensi yang tinggi.
2. *Setting* parameter yang tidak baik.
3. Kerusakan *hardware*.
4. Area cakupan radio yang tidak baik.
5. *Neighbouring cell relation* yang tidak perlu.
6. Masalah antena *receiver* atau *hardware* BTS.

## 2.3 Jaringan Komunikasi *Mobile*

Menggunakan jaringan komunikasi *mobile* umum yang sudah ada untuk *backhauling* merupakan alternatif yang tepat untuk menyediakan koneksi *backhaul* antara kereta api yang bergerak dengan tanah. Jaringan komunikasi *mobile* yang akan digunakan adalah sebagai berikut.

### 2.3.1 *General Packet Radio Service (GPRS)*

GPRS merupakan sistem transmisi berbasis paket untuk *Global System for Mobile Communication (GSM)* yang menggunakan prinsip *tunnelling*. GPRS juga dikenal sebagai teknologi 2,5 G yang merupakan teknologi peralihan antara 2G dan 3G. GPRS menawarkan *bitrate* yang lebih tinggi jika dibandingkan dengan penggunaan teknologi *Circuit Switch Data* atau CSD. *Bitrate*-nya secara kasar dapat

mencapai 160 kbit/s, jauh lebih cepat jika dibandingkan dengan 9,6 kbit/s yang dapat dicapai oleh rangkaian GSM *switched*. Kanal-kanal radio ganda dapat dialokasikan bagi seorang pengguna dan kanal yang sama dapat pula digunakan secara berbagi (*sharing*) di antara beberapa pengguna sehingga menjadi sangat efisien.

Dari segi biaya, pentarifan diharapkan hanya mengacu kepada volume penggunaan. Pengguna hanya dikenakan biaya berdasarkan banyaknya *byte* yang dikirim atau diterima, tanpa mempedulikan panggilan. Dengan demikian, pelanggan akan lebih cenderung untuk mengakses GPRS dibandingkan dengan layanan-layanan IP.

Berdasarkan teori, GPRS menjanjikan *bitrate* mulai dari 56 kbit/s sampai 115 kbit/s sehingga memungkinkan akses internet, pengiriman data multimedia ke komputer, *notebook*, dan *handheld computer* [2]. Namun, dalam implementasinya hal tersebut sangat tergantung pada faktor-faktor sebagai berikut:

1. Konfigurasi dan alokasi *timeslot* pada level BTS.
2. *Software* yang digunakan.
3. Dukungan fitur dan aplikasi ponsel yang digunakan.

Hal ini menjelaskan mengapa pada saat-saat tertentu dan di lokasi tertentu akses GPRS terasa lambat, bahkan lebih lambat dari akses CSD yang memiliki *bitrate* 9,6 kbit/s.

Jaringan GPRS memiliki beberapa komponen utama serta fungsinya sebagai berikut:

1. *Gateway GPRS Support Node* (GGSN): Gerbang penghubung jaringan GPRS ke jaringan internet. Fungsi dari komponen ini adalah sebagai *interface* ke *Public Data Network* (PDN), *information routing*, *network screening*, *user screening*, dan *address mapping*.
2. *Serving GPRS Support Node* (SGSN): Gerbang penghubung jaringan BSS/BTS ke jaringan GPRS. Komponen ini berfungsi untuk mengantarkan paket data ke MS, *update* pelanggan ke HLR, dan registrasi pelanggan baru.

3. PCU: Komponen di level BSS yang menghubungkan terminal ke jaringan GPRS.

GPRS menggunakan sistem komunikasi *packet switch* sebagai cara untuk mentransmisikan datanya. *Packet switch* adalah sebuah sistem dimana data yang akan ditransmisikan dibagi menjadi bagian-bagian kecil (paket) lalu ditransmisikan dan diubah kembali menjadi data semula. Sistem ini dapat mentransmisikan ribuan bahkan jutaan paket per detik. Transmisi dilakukan melalui PLMN (*Public Land Mobile Network*) dengan menggunakan *IP backbone*. Karena memungkinkan untuk pemakaian kanal transmisi secara bersamaan oleh pengguna lain, maka biaya akses GPRS secara teori lebih murah daripada biaya akses CSD.

GPRS didesain untuk menyediakan layanan transfer paket data pada jaringan GSM dengan kecepatan yang lebih baik dari GSM. Kecepatan yang lebih baik ini didapat dengan menggunakan *Coding Scheme* (CS) yang berbeda dari GSM.

### **2.3.2 Enhanced Data rates for GSM Evolution (EDGE)**

EDGE merupakan pengembangan dari jaringan GSM yang didesain untuk membagi sumber daya kanal radio secara dinamis antara layanan *packet service* dengan layanan *circuit switch* GSM. Standar EDGE menawarkan akses berbasis *packet switch* dimana sumber daya kanal fisik yang ada akan dibagi secara efisien antara pemakai yang sedang aktif. Kanal frekuensi yang ada diberikan kepada pelanggan hanya ketika diperlukan oleh pengguna. Dengan menggunakan teknologi ini, sejumlah pengguna akan membagi kanal radio dengan mengadaptasikan kecepatan data masing-masing sehingga kecepatan data yang tinggi akan diperoleh ketika banyak sumber daya yang sedang tidak digunakan.

EDGE memberikan akses *bitrate* mencapai 473.6 kbit/s, tiga kali lebih cepat jika dibandingkan dengan generasi sebelumnya (GPRS) dalam hal pengiriman data secara paket [3]. Selain itu EDGE sangat mudah diimplementasikan sehingga operator tidak perlu membangun jaringan baru yang membutuhkan biaya yang sangat besar. Hal ini dikarenakan EDGE hanya memperkenalkan teknik modulasi 8-PSK.

### 2.3.3 *Universal Mobile Telecommunication System (UMTS)*

UMTS merupakan standar sistem telekomunikasi generasi ketiga (3G) berbasis WCDMA yang dibangun di atas platform GSM dengan kecepatan akses mencapai 2 Mbit/s [4]. UMTS-WCDMA dan GSM dapat digelar bersama dengan memanfaatkan jaringan inti (*Core Network*) yang sama. Secara umum, ada tiga bagian penting dari arsitektur sistem teknologi UMTS, yaitu:

1. *User Equipment (UE)*

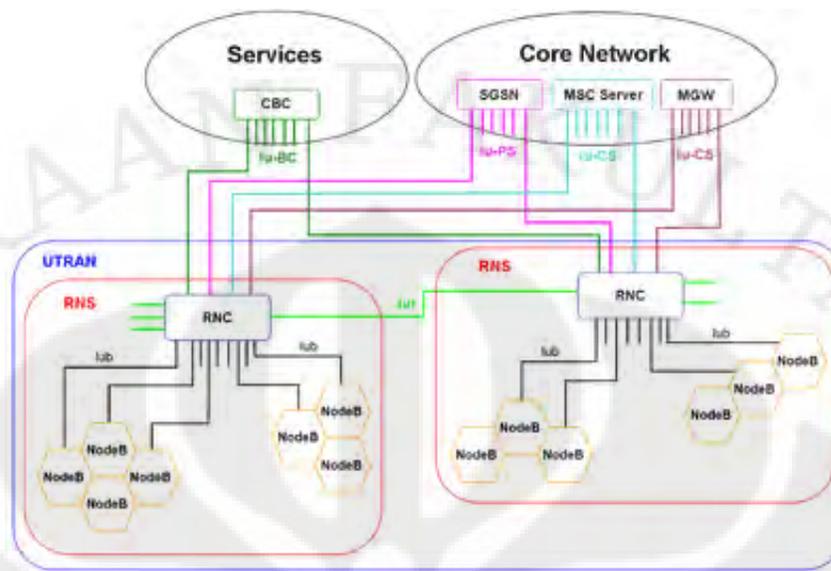
*User Equipment (UE)* atau *Mobile User (ME)* merupakan ponsel dan *Subscriber Identity Module (SIM) Card* yang disebut dengan *Universal SIM (USIM)*. USIM berisikan data spesifik pelanggan dan berfungsi untuk keperluan otentikasi pelanggan agar dapat masuk ke jaringan.

2. *UMTS Radio Access Network (UTRAN)*

*Link* radio antara UTRAN dan UE disebut *UMTS Radio Access (UTRA)*. UTRAN merupakan kumpulan beberapa RNS yang terdiri dari RNC, dan *Node B*. Setiap *node-node* tersebut selalu dihubungkan dengan masing-masing *interface* yang berbeda. *Interface* antara *Core Network* dengan RNC adalah Iups/Iucs, sesama RNC menggunakan Iur, antarRNC dengan *Node B* menggunakan Iub, dan dari UTRAN ke UE menggunakan Uu.

3. *Core Network (CN)*

*Core Network* atau jaringan inti terdiri dari *Circuit Switch Domain (CS Domain)*, dan *Packet Switch (PS)*. *Circuit Switch* terdiri dari *Mobile Switching Centre (MSC) Server*, *Visitor Location Register (VLR)*, dan *Gateway MSC (GMSC)*. *Packet Switch* terdiri dari *Serving GPRS Support Node (SGSN)*, dan *Gateway GPRS Support Node (GGSN)*, *Firewall*, *Border Gateway (BG)*, *DNS Server*, dan *Charging Gateway*. Beberapa elemen jaringan seperti *Equipment Identity Register (EIR)*, *Home Location Register (HLR)*, dan *Authentication Centre (AuC)* merupakan bagian dari keduanya.



Gambar 2.2 Arsitektur jaringan UMTS [4]

#### 2.3.4 High-Speed Downlink Packet Access (HSDPA)

HSDPA adalah sebuah protokol telepon genggam dan kadang disebut sebagai teknologi 3,5G. HSDPA fase pertama berkapasitas 4,1 Mbps. Kemudian menyusul fase 2 berkapasitas 11 Mbps dan kapasitas maksimal *downlink peak data rate* hingga mencapai 14 Mbit/s [5]. Teknologi ini dikembangkan dari W-CDMA sama seperti EV-DO mengembangkan CDMA2000. HSDPA memberikan jalur evolusi untuk jaringan UMTS yang memungkinkan untuk penggunaan kapasitas data yang lebih besar (sampai 14,4 Mbit/s *downlink*).

#### 2.4 Kualitas Layanan (QoS)

Kemampuan menyediakan jaminan performansi dan diferensiasi layanan dalam jaringan sering diacu dengan istilah QoS (*quality of service*). ITU, dalam rekomendasi E.800, mendefinisikan QoS sebagai pengaruh kolektif atas performansi layanan yang menentukan tingkat kepuasan pengguna layanan. QoS-Forum mendefinisikan QoS sebagai ukuran kolektif atas tingkat layanan yang disampaikan ke pelanggan, ditandai dengan beberapa kriteria yang meliputi *availabilitas*, *error*

*performance*, *response time* dan *throughput*, sambungan atau transmisi yang hilang akibat kongesti, waktu *setup*, dan kecepatan deteksi dan koreksi kesalahan.

#### 2.4.1 Performansi

Performansi jaringan merujuk ke tingkat kecepatan dan keandalan penyampaian berbagai jenis beban data di dalam suatu sistem komunikasi. Performansi merupakan kumpulan berbagai besaran teknis, antara lain:

1. *Availabilitas*, yaitu persentase hidupnya sistem atau sub-sistem telekomunikasi. Idealnya, *availabilitas* harus mencapai 100%. Nilai *availabilitas* yang diakui cukup baik adalah 99,9999% (*six nines*), yang menunjukkan tingkat kerusakan sebesar 2,6 detik per bulan [6].
2. *Throughput*, yaitu kecepatan (*rate*) transfer data efektif, yang diukur dalam bit/s. *Header-header* dalam paket-paket data mengurangi nilai *throughput*. Maka penggunaan sebuah saluran secara bersama-sama juga akan mengurangi nilai ini.
3. *Packet loss*, adalah jumlah paket yang hilang. Umumnya perangkat jaringan memiliki *buffer* untuk menampung data yang diterima. Jika terjadi kongesti yang cukup lama, *buffer* akan penuh, dan data baru tidak diterima. Paket yang hilang ini harus diretransmisi, yang akan membutuhkan waktu tambahan. Umumnya nilai *packet loss* diharuskan kurang dari 1%, dalam waktu misalnya satu bulan.
4. *Latency*, adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* ini bisa dipengaruhi oleh jarak (misalnya akibat pemakaian satelit), atau kongesti (yang memperpanjang antrean), atau bisa juga akibat waktu olah yang lama (misalnya untuk *digitizing* dan kompresi data).
5. *Jitter*, atau variasi dalam *latency*, diakibatkan oleh variasi-variasi dalam panjang antrean, dalam waktu pengolahan data, dalam waktu yang dibutuhkan untuk retransmisi data (karena jalur yang digunakan juga berbeda), dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan.

### 2.4.2 Diferensiasi

Berbagai aplikasi memiliki jenis kebutuhan yang berbeda. Misalnya transfer data bersifat sensitif terhadap distorsi tetapi kurang sensitif terhadap *delay*. Sebaliknya, komunikasi suara bersifat sensitif terhadap *delay* dan kurang sensitif terhadap distorsi. Tabel berikut memaparkan tingkat kepekaan performansi yang berbeda untuk jenis layanan jaringan yang berlainan.

Tabel 2.1 Kepekaan performansi untuk jenis layanan jaringan [6]

LAYANAN	KEPEKAAN PERFORMANSI			
	BAND WIDTH	LOSS	DELAY	JITTER
Voice	Rendah	Medium	Tinggi	Tinggi
Transaksi Data	Rendah	Tinggi	Tinggi	Rendah
Email	Rendah	Tinggi	Rendah	Rendah
Browsing Biasa	Rendah	Medium	Medium	Rendah
Browsing Serius	Medium	Tinggi	Tinggi	Rendah
Transfer File	Tinggi	Medium	Rendah	Rendah
Video Conference	Rendah	Medium	Tinggi	Tinggi
Multicasting	Tinggi	Tinggi	Tinggi	Tinggi

### 2.4.3 QoS dalam IP

IP tidak memiliki mekanisme pemeliharaan QoS. Protokol seperti TCP memang memungkinkan jaminan validitas data, sehingga *suite* TCP/IP selama ini dianggap cukup ideal bagi transfer data. Tetapi verifikasi data mengakibatkan *delay* hantaran paket. Lagipula mekanisme ini tidak dapat digunakan untuk paket dengan protokol UDP, seperti suara dan video.

Beberapa skema telah diajukan untuk mengelola QoS dalam jaringan IP. Dua skema utama adalah *Integrated Services (IntServ)* dan *Differentiated Services (DiffServ)*. *IntServ* bertujuan menyediakan sumber daya seperti *bandwidth* untuk *traffic* dari ujung ke ujung. Sementara *DiffServ* bertujuan membagi *traffic* atas kelas-kelas yang kemudian diberi perlakuan yang berbeda.

## 2.5 Performansi Jaringan Komunikasi *Mobile*

Penggunaan jaringan komunikasi *mobile* umum yang sudah ada untuk *backhauling* seperti yang telah disebutkan sebelumnya memang merupakan alternatif yang tepat. Namun, terdapat sejumlah kekurangan sebagai berikut:

1. Dibandingkan dengan yang lain, penggunaan HSDPA merupakan pilihan yang tepat. Namun, dalam hal cakupan nasional untuk akses nirkabel *broadband* masih belum tersedia dan belum tentu tersedia di tahun-tahun berikutnya, tergantung kepada Negara dan operator.
2. Penggunaan jaringan komunikasi data *mobile* umum dalam kondisi *roaming* internasional dapat meningkatkan biaya konektivitas *backhaul* ke level yang tidak dapat diterima.
3. Jaringan yang mempunyai cakupan yang bagus (seperti GPRS atau EDGE) mempunyai kelemahan lain, seperti *bandwidth* yang terlalu terbatas dan tidak mendukung mobilitas yang cepat dari terminal. Selain itu keduanya mempunyai *packet delay* yang tinggi dan bervariasi sehingga menimbulkan permasalahan pada *handover* yang cepat.

Berdasarkan pengujian yang telah dilakukan Siemens Networks, IBBT, PATS, dan IBCN research group di Belgia pada tahun 2006, diperoleh hasil sebagai berikut.

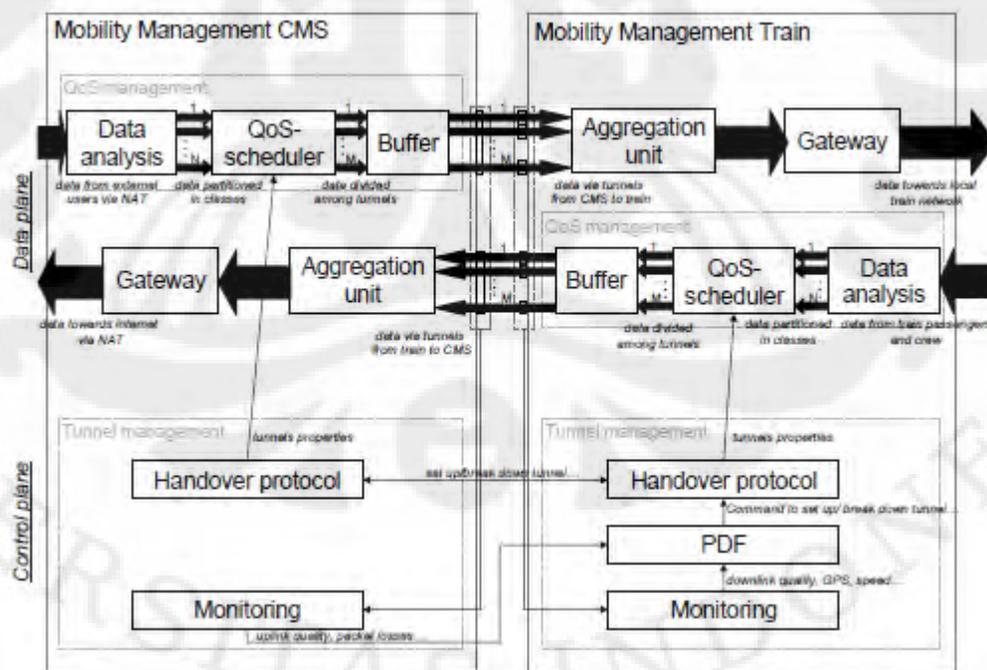
Tabel 2.2 Perbandingan hasil pengujian GPRS, EDGE, UMTS, dan HSDPA [7]

	speed [km/h]	Throughput [kbit/s]	RTT [ms]			handover delay [s]
			Min	Avg.	Max	
GPRS	0	41,6	484	648	2337	5 - 10
	120	29,6	484	648	2337	5 - 10
EDGE	0	170	290	683	2993	5 - 10
	120	28	290	683	2993	5 - 10
UMTS	0	360	NA	160	NA	< 2
	120	324 - 360	NA	160	NA	< 2
HSDPA	0	1000 peak	NA	140	NA	NA
	120	1000 peak	NA	140	NA	NA

Hasil pengujian tersebut menunjukkan bahwa *handover* antarsistem dapat menimbulkan *packet loss* dan *delay*, yang akan memberikan dampak pada QoS dari aplikasi penumpang. Oleh karena itu, agar layanan internet dapat tersedia di kereta api dibutuhkan suatu arsitektur yang menggabungkan akses *broadband*, skalabilitas, *handover* yang mulus, dan jaminan kualitas layanan dalam satu solusi.

## 2.6 Arsitektur

Penumpang cenderung mencoba untuk menjaga koneksi yang mereka miliki, baik dengan menggunakan teknologi ganda maupun tunggal. Selanjutnya, jika setiap penumpang ingin menjaga koneksi internet oleh dirinya sendiri (misalnya dengan UMTS/GPRS PCMCIA card), *base station* harus menangani pengguna *co-located* dalam jumlah besar dan akibatnya banyak terjadi *handover* secara simultan. Teknologi jaringan *broadband* yang ada saat ini tidak dapat menyediakan QoS yang diperlukan pada kondisi tersebut. Selain itu, untuk menangani degradasi sinyal di dalam gerbong, *repeater* harus diinstalasikan pada kereta api [7].



Gambar 2.3 Arsitektur: gateway pada central dan kereta api [7]

Universitas Indonesia

Oleh karena itu, suatu solusi ditawarkan dimana penumpang terhubung ke internet dengan menggunakan *on-board WiFi access point* yang terhubung ke jaringan kereta lokal. Suatu sistem *gateway* yang cerdas juga terhubung ke *backbone on-board* pada kereta api, menjumlahkan seluruh *backhaul* koneksi nirkabel yang tersedia terhadap *Central Management System*, yang terhubung ke *mainland*. Dengan begitu, kereta api dapat berganti dari satu akses jaringan ke akses jaringan yang lain. Pergantian ini harus dilakukan secara transparan, dimana penumpang tidak boleh mengalami gangguan koneksi dan sejenisnya. Komponen dari *gateway* pada kereta api dan *Central Management System* ditunjukkan oleh gambar 2.3 dan akan dibahas lebih mendalam pada paragraf berikut [7].

### 2.6.1 *Policy Decision Function*

*Policy Decision Function* (PDF) merupakan suatu komponen arsitektur yang sangat penting. Tugasnya adalah menentukan *interface* yang akan digunakan untuk membuat koneksi antara kereta api dan jaringan akses yang tepat. PDF dapat membuat keputusan tersebut berdasarkan informasi dari *monitoring unit*, yang memberikan data mengenai kualitas *link*, letak dan kecepatan kereta api, posisi kereta api, dan halangan *line of sight* terhadap satelit. Selain itu, parameter lain seperti biaya atau *load balancing* dapat dimasukkan ke dalam perhitungan. Komponen ini mempunyai kemampuan untuk membedakan teknologi jaringan dan berperan sebagai suatu *abstraction layer* bagi komponen arsitektur jaringan internet di kereta api yang lain. Protokol *Mobility Management* akan memperoleh input dari PDF agar dapat bertukar pesan pada waktu yang tepat untuk melakukan *vertical handover* [7].

### 2.6.2 *Mobility Management*

Modul *Mobility Management* sebagian terletak di kereta api dan sebagian lagi terletak di *Central Management System*. Modul tersebut melakukan seluruh tugas mobilitas tertentu agar *handover* dapat dilakukan semulus mungkin. Untuk protokol *handover Mobility Management* akan digunakan *Mobile IP* (MIP). Protokol tersebut

akan menganggap kereta api sebagai *mobile router* dan penumpang sebagai *mobile network*. Beberapa ekstensi dan optimasi untuk MIP telah diimplementasikan agar protokol tersebut dapat beradaptasi dengan lingkungan kereta api dan untuk berperan dalam mekanisme *encapsulation*. Ditekankan bahwa dalam kasus ini, pengguna tidak akan memerlukan perangkat yang *aware* terhadap MIP untuk mengakses jaringan: protokol *handover* tersebut akan meng-*encapsulate* IP traffic dari seluruh pengguna di kereta api [7].

## 2.7 Tinjauan Mobile IP (MIP)

Mobile IP (MIP), yang juga dikenal dengan nama *IP Mobility Support for IPv4*, pertama kali didefinisikan di *Request for Comments* (RFC) 2002, diperbarui di RFC 3220, dan sekarang dijelaskan di RFC 3344. MIP memiliki atribut kunci dan fitur-fitur sebagai berikut [8]:

1. *Seamless device mobility using existing device address*: Perangkat *mobile* dapat mengubah lokasi dan metode *network attachment* fisiknya sambil tetap menggunakan *IP address* yang ada.
2. *No new addressing or routing requirements*: Keseluruhan skema pengalamatan dan *routing* seperti pada IP biasa dipertahankan. *IP address* masih diberikan dengan cara konvensional oleh pemilik masing-masing perangkat. Tidak ada persyaratan *routing* baru yang ditempatkan di *internetwork*, seperti rute *host* tertentu.
3. *Interoperability*: Perangkat MIP masih dapat mengirim dan menerima dari perangkat IP yang ada, yang tidak tahu bagaimana MIP bekerja, dan sebaliknya.
4. *Layer transparency*: Perubahan yang dibuat oleh MIP terbatas pada lapisan *network*. Lapisan *transport* dan protokol lapisan yang lebih tinggi dan aplikasi dapat berfungsi seperti pada IPv4 biasa, dan koneksi yang ada bahkan dapat dijaga ketika sedang berpindah.

5. *Limited hardware changes*: Perubahan diperlukan pada *software* perangkat *mobile*, demikian juga *router* yang digunakan secara langsung oleh perangkat *mobile*. Namun, perangkat yang lain tidak memerlukan perubahan, termasuk *router* antara *home* dan *foreign network*.
6. *Scalability*: MIP memungkinkan suatu perangkat untuk berganti dari satu jaringan ke jaringan yang lain, dan mendukung hal tersebut untuk jumlah perangkat yang berubah-ubah. Cakupan perubahan koneksi dapat bersifat global; Kita dapat men-*detach notebook* dari kantor di London dan membawanya ke Australia atau Brazil, sebagai contoh, dan *notebook* tersebut akan bekerja sama seperti jika kita membawanya ke kantor sebelah.
7. *Security*: MIP bekerja dengan mengalihkan pesan, termasuk prosedur otentikasi untuk mencegah perangkat asing yang dapat menyebabkan masalah.

Dengan MIP, *mobile node* bisa mempunyai dua buah *address*, yaitu [8]:

1. *Home address*: IP permanen yang diberikan untuk *mobile node*. *Address* ini digunakan oleh perangkat pada *home network*-nya dan merupakan *address* dimana *datagram* yang ditujukan untuk *mobile node* selalu dikirim.
2. *Care of address (CoA)*: *Address* sementara yang digunakan oleh *mobile node* ketika sedang “bepergian” jauh dari *home network*-nya. CoA merupakan IP 32-bit biasa dalam banyak hal, namun hanya digunakan oleh MIP untuk meneruskan *datagram* IP dan untuk fungsi administratif. Lapisan yang lebih tinggi tidak pernah menggunakan CoA, begitu juga dengan perangkat IP biasa ketika membuat *datagram*.

Adanya CoA inilah yang memungkinkan *mobile node* untuk mengubah *point of attachment*-nya ke internet tanpa perlu mengubah IP *address*.

### 2.7.1 Prinsip Kerja Mobile IP (MIP)

Prinsip kerja MIP secara umum dapat digambarkan dalam beberapa proses sederhana sebagai berikut [8]:

1. *Agent communication*: *Mobile node* menemukan *agent* pada jaringan lokalnya dengan melibatkan diri dalam proses *agent discovery*. *Mobile node* “mendengarkan” pesan *agent advertisement* yang dikirim oleh *agent* dan dari pesan ini dapat ditentukan di mana *mobile node* berada. Jika *mobile node* tidak “mendengar” pesan tersebut, *mobile node* dapat meminta untuk menggunakan pesan *agent solicitation*.
2. *Network location determination*: *Mobile node* menentukan apakah sedang berada di *home* atau *foreign network* dengan melihat informasi pada pesan *agent advertisement*.

Jika *mobile node* sedang berada di *home network*, maka yang digunakan adalah IP biasa. Namun, jika *mobile node* berpindah ke *foreign network*, maka proses selanjutnya adalah sebagai berikut:

3. *Care of address acquisition*: Perangkat memperoleh suatu *address* sementara yang disebut *care of address*. *Address* ini bisa berasal dari pesan *agent advertisement* dari *foreign agent*, atau melalui cara lain. *Address* ini digunakan hanya sebagai *destination point* untuk meneruskan *datagram*, dan bukan untuk tujuan lain.
4. *Agent registration*: *Mobile node* menginformasikan *home agent* pada *home network* mengenai keberadaannya pada *foreign network* dan memungkinkan penerusan *datagram* dengan mendaftarkan *home agent*. Hal ini dapat dilakukan secara langsung antara *node* dan *home agent*, atau secara tidak langsung menggunakan *foreign agent* sebagai perantara.
5. *Datagram forwarding*: *Home agent* meng-capture *datagram* yang ditujukan untuk *mobile node* dan meneruskannya. *Datagram* dapat dikirim secara langsung ke *node* atau secara tidak langsung ke *foreign agent* untuk diantarkan, tergantung pada jenis CoA yang digunakan.

*Datagram forwarding* terus berlangsung sampai *agent registration* saat itu berakhir. Perangkat kemudian dapat memperbaruinya kembali. Jika perangkat berpindah lagi,

proses untuk memperoleh CoA akan diulang kembali dan mendaftarkan lokasi barunya dengan *home agent*. Ketika *mobile node* kembali ke *home network*-nya, *mobile node* akan melakukan deregistrasi untuk membatalkan *datagram forwarding* dan kembali menggunakan IP biasa.

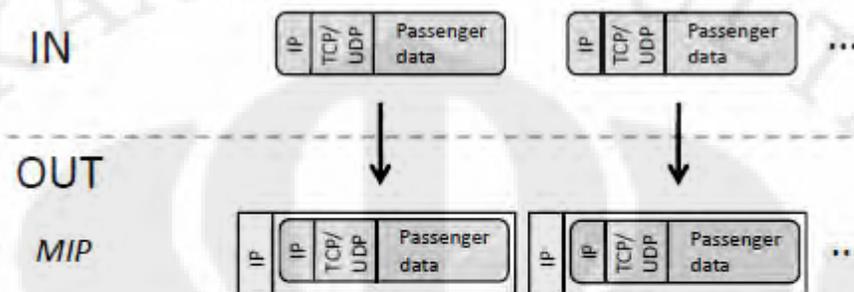
Untuk di kereta api, proses yang terjadi adalah sebagai berikut. *Mobile node* akan selalu teridentifikasi melalui *home address*-nya, tanpa mepedulikan lokasinya. *Home agent*, yang terletak di *home network* menerima seluruh paket yang ditujukan untuk *mobile node* dan meneruskannya melalui suatu IP *tunnel* menuju *mobile node*. Melalui registrasi, *home agent* selalu dapat mengetahui lokasi *mobile node*. Paket yang dikirimkan oleh penumpang akan di-*capture* oleh *access router* kereta api dan di-*encapsulate* dalam sebuah IP *header* menuju *home agent* yang berada pada *Central Management System*. Seperti yang telah disebutkan sebelumnya, *mobile node* sebagai contoh, bukan merupakan *endpoint* dari koneksi. Akan tetapi, *mobile node* dianggap sebagai *router* dengan *mobile network* penumpang. Hal ini menyebabkan pengguna tidak perlu menjaga koneksi MIP mereka. Selanjutnya, *router advertisement* dihilangkan dalam arsitektur, sebab PDF menyediakan *Mobility Management* dengan informasi yang diperlukan mengenai akses jaringan.

Kemudian fokus diarahkan dalam meminimalkan *actual handover delay* dengan melakukan MIP *pre-registration* segera setelah komponen *Mobility Management* mengetahui bahwa *handover* sudah dekat. Segera setelah PDF memberitahukan bahwa *handover* akan segera terjadi, komponen *Mobility Management* mengirimkan permintaan registrasi MIP ke *Central Management System* tepat setelah proses pergantian selesai.

### 2.7.2 Overhead Pada Mobile IP (MIP)

MIP melibatkan beberapa *overhead* untuk menjamin mobilitas yang transparan bagi pengguna. Berikut ini adalah hasil secara teori mengenai *overhead* pada MIP. Data yang dikirimkan oleh penumpang dikumpulkan dalam suatu *header*

TCP (20 *bytes*) atau UDP (8 *bytes*), ditambah dengan *header* IP (20 *bytes*). Paket-paket tersebut kemudian akan di-*capture* oleh *gateway* kereta api.



Gambar 2.4 MIP encapsulation [7]

Pada *gateway* kereta api, MIP akan meng-*encapsulate* paket tersebut dalam suatu IP *header* dengan *home agent* sebagai alamat tujuan [7].

## 2.8 Efek Doppler

Pergerakan *mobile node* relatif terhadap *base station* akan menyebabkan terjadinya pergeseran frekuensi *bandwidth* yang disebut dengan efek *Doppler*. Efek *Doppler* akan menyebabkan perbedaan *packet delay* pada *mobile node*. Besarnya efek *Doppler* tersebut ditentukan oleh kecepatan *mobile node* relatif terhadap *base station*.

## **BAB III PERANCANGAN MODEL SIMULASI**

### **3.1 Gambaran Umum Simulasi**

Pada skripsi ini akan dibuat suatu simulasi topologi jaringan *wired-cum-wireless* (jaringan yang merupakan kombinasi dari jaringan kabel dan jaringan nirkabel) yang bertujuan untuk mengamati bagaimana paket data diteruskan dari *home agent* ke *foreign agent* sesuai dengan fungsi dari *Mobile IP* (MIP) untuk diaplikasikan pada jaringan internet di kereta api.

Simulasi dibuat dengan menggunakan *software* Network Simulator 2.29 yang dijalankan melalui VMware 6.5.0 dengan sistem operasi Windows Vista Ultimate *Service Pack 2*. Simulasi dijalankan di atas *notebook* dengan spesifikasi sebagai berikut:

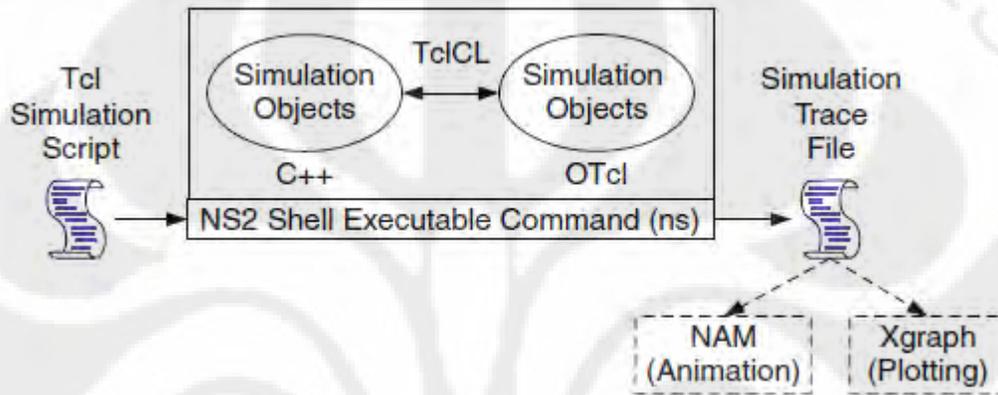
1. Prosesor Intel Core2 Duo T5500 1,66 GHz.
2. RAM Kingston DDRII 2,5 GB.
3. Harddisk Fujitsu 120 GB.
4. Semua aksesori yang terintegrasi pada *notebook* BenQ Joybook S31V.

### **3.2 Network Simulator**

Network Simulator merupakan simulator jaringan yang berorientasi objek dan dikendalikan oleh *event* atau kejadian yang dikondisikan sebelumnya. Simulasi fungsi dan protokol jaringan kabel dan nirkabel seperti algoritma *routing*, *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP), dan variannya dapat dilakukan dengan menggunakan Network Simulator.

Network Simulator dikembangkan dalam dua bahasa, yaitu C++ dan *Object-oriented Tool Command Language* (Otcl), dimana keduanya dapat menyusun simulasi yang akan dilakukan atau membuat suatu modul jaringan yang baru. Sifat simulator lebih seperti *interpreter* dari *source code* C++ atau Otcl sehingga ada suatu

kebebasan dan kemudahan dalam membuat suatu simulasi yang unik dan spesifik. Selain itu, Network Simulator juga memiliki Event Scheduler sebagai pengatur jalannya simulasi dan komponen *library* yang menyediakan berbagai objek simulasi serta *network setup* atau *plumbing module* yang berfungsi menjembatani modul-modul dan objek-objek yang digunakan dalam simulasi.



Gambar 3.1 Arsitektur dasar Network Simulator [9]

Sebagai input dari simulator, digunakan *source code* dalam bahasa C++ atau Otcl yang berisi kejadian-kejadian pada simulasi dan objek simulator yang digunakan. Kemudian, input akan dijalankan oleh simulator melalui suatu scheduler dan akan mengeluarkan hasil yang dapat berupa *trace file*, statistik, grafik, atau animasi dalam format *.nam* yang dapat diamati dengan menggunakan Network Animator (NAM) [9].

### 3.2.1 Event Scheduler

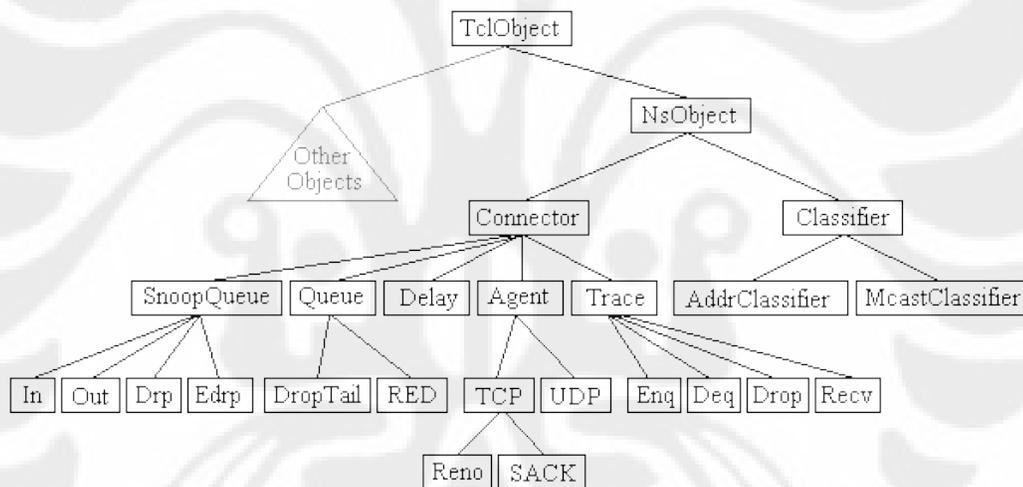
Salah satu komponen utama pada Network Simulator adalah Event Scheduler yang bertindak sebagai pengatur komponen jaringan yang akan menentukan suatu *delay* atau pengatur waktu dari kejadian yang direncanakan. Ada dua jenis scheduler yang digunakan, yaitu *real-time* dan *non-real-time*. *Real-time* scheduler digunakan untuk emulasi yang memungkinkan simulator berinteraksi dengan jaringan yang

sesungguhnya. Sedangkan *non-real-time* scheduler memiliki tiga macam penggunaan, yaitu sebagai *List*, *Heap*, dan *Calendar*.

Penggunaan utama dari Event Scheduler adalah sebagai penjadwal kejadian pada simulasi yang sedang berjalan, seperti mengatur kapan FTP mulai berjalan, kapan simulasi berakhir, atau mulai menjalankan *simulation trace* [9].

### 3.2.2 Komponen Jaringan

Sebagai peniru dari jaringan yang sesungguhnya, komponen dari jaringan yang sebenarnya haruslah menjadi bagian dari suatu Network Simulator. Berikut ini adalah hirarki dari beberapa komponen jaringan pada Network Simulator.



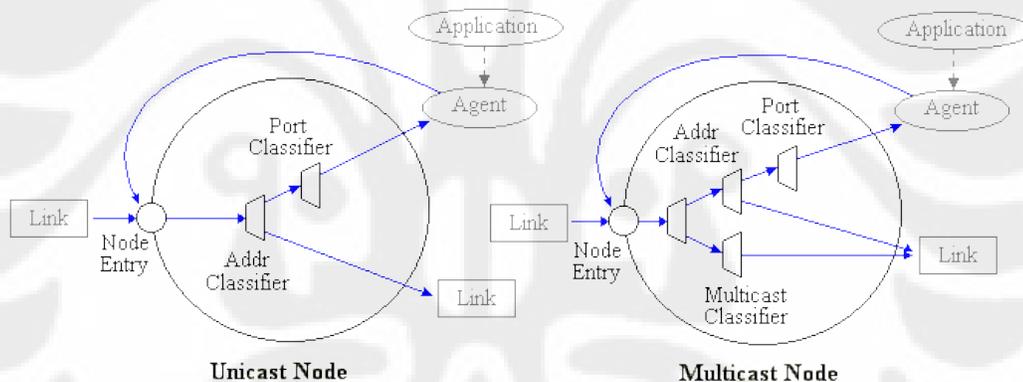
Gambar 3.2 Hirarki komponen jaringan pada Network Simulator [9]

Puncak dari hirarki tersebut adalah *TclObject class* yang menjadi *superclass* untuk seluruh *library object* seperti scheduler, komponen jaringan, *timer*, dan sebagainya. Sedangkan *NsObject class* bertindak sebagai *superclass* dari semua objek komponen dasar dari jaringan yang menangani paket, yang bisa terdiri dari gabungan objek jaringan seperti *node* dan *link*. Komponen dasar dari jaringan dibagi menjadi dua sub-kelas, yaitu *Connector* dan *Classifier*. Objek komponen yang hanya memiliki

satu jalur output data berada di bawah kelas *Connector*, sementara yang memiliki jalur output data yang banyak berada di bawah kelas *Classifier* [9].

### 3.2.2.1 Node

*Node* adalah objek gabungan yang terdiri dari objek *entry node* dan *classifier*. Ada dua macam *node* pada Network Simulator, yaitu *unicast* dan *multicast*. *Unicast node* adalah *address classifier* yang melakukan *unicast routing* dan memiliki *port classifier*. Sedangkan *multicast node* akan melakukan *multicast routing* dan klasifikasi dengan *multicast classifier*. *Node* yang menjadi standar pada Network Simulator adalah *unicast node*, sedangkan *multicast node* diperoleh melalui input script Otcl [9].



Gambar 3.3 Unicast dan Multicast node [9]

### 3.2.2.2 Link

Komponen ini sangat penting dalam penyusunan suatu simulasi. Di sini paket akan mengalami *delay*, *dropping*, dan dapat dimonitor dengan fasilitas *tracing*. Selain menghubungkan *node* yang satu dengan yang lain, *link* menjadi tempat lewatnya paket-paket dan tempat membuang paket yang mengalami *overflow* atau terkena *drop*. Sifat *link* dapat diatur sebagai *simplex* dan *duplex* [9].

### 3.2.2.3 Antrean

Ada beberapa macam antrean yang disediakan oleh Network Simulator, seperti *DropTail*, CBQ, RED, FQ, dan sebagainya. Komponen ini akan menempatkan paket yang datang ke dalam suatu *buffer* sebelum akhirnya diteruskan ke dalam *link*. Umumnya parameter utama dari komponen ini adalah ukuran *buffer*. Sedangkan parameter yang lain sangat tergantung kepada jenis antrean [9].

### 3.2.2.4 Agent

Sebagai pembawa informasi di dalam jaringan atau paket yang beredar, maka peranan *agent* sangatlah penting dalam suatu simulasi. Di sini *agent* bertindak seperti protokol *layer 4* yang bertugas mengatur pengiriman paket secara *end-to-end*. Sumber dan *sink* bertindak sebagai kesatuan fungsi *layer 4*. *Agent* yang banyak digunakan antara lain UDP dan TCP dengan bermacam-macam variannya. Parameter utamanya antara lain *packet size* dan *flow ID* [9].

### 3.2.2.5 Aplikasi

Sebagai pemeran fungsi *layer atas*, beberapa bentuk aplikasi disediakan pada Network Simulator. Aplikasi ini akan menumpang di atas *agent* sehingga suatu *agent* perlu ditempelkan pada aplikasi agar paket-paketnya dapat terkirim. Parameter utama yang biasanya ditentukan antara lain adalah besarnya *rate* pengiriman [9].

### 3.2.2.6 Tracing

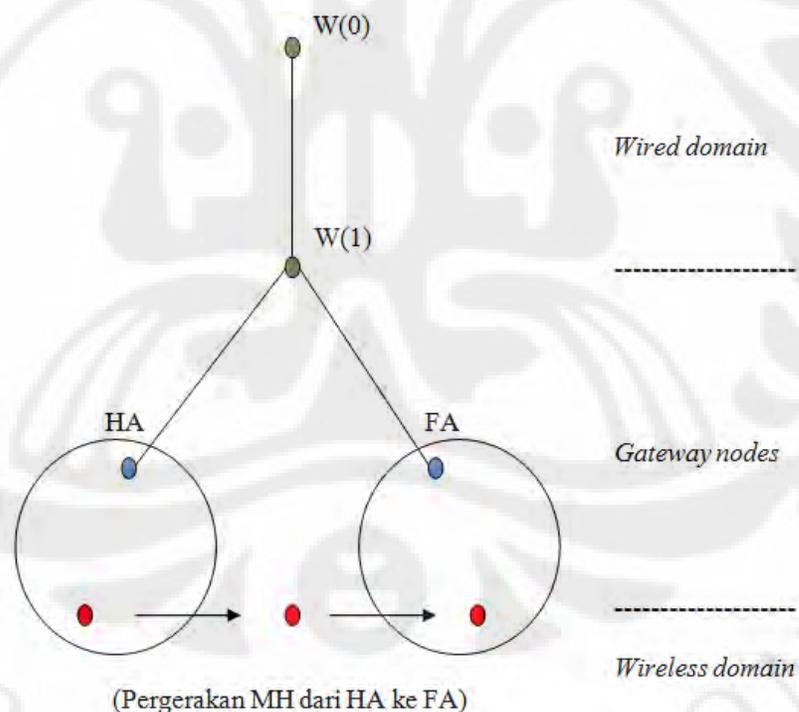
Untuk dapat mengetahui hasil simulasi, maka setiap kejadian dalam simulasi perlu diikuti dan dicatat. Hal ini biasanya dilakukan untuk suatu parameter tertentu seperti *bandwidth* atau *packet loss*. Berbagai macam bentuk output dapat dihasilkan, mulai dari hasil numerik, grafis, hingga animasi. Hasil *trace* yang paling sering digunakan adalah *trace file* dalam format *.tr* dan animasi dari simulasi dalam format *.nam* [9].

### 3.3 Perancangan Simulasi

Simulasi topologi jaringan *wired-cum-wireless* yang akan dibuat terdiri dari beberapa komponen sebagai berikut:

1. Dua buah *wired node* (W0 dan W1), yang masing-masing bertindak sebagai *server* dan *impairment node*.
2. Dua buah *base station node* (*home agent* dan *foreign agent*), yang bertindak sebagai *gateway* antara dua domain untuk menukar paket di antara *wired* dan *wireless node*.
3. Satu buah *mobile node*, yang akan berpindah antara *home agent* dan *foreign agent*, yang dinamakan dengan *Mobile Host* (MH).

Pada topologi ini, W1 terhubung dengan *home agent* dan *foreign agent* seperti yang ditunjukkan pada gambar berikut.



Gambar 3.4 Topologi jaringan *wired-cum-wireless*

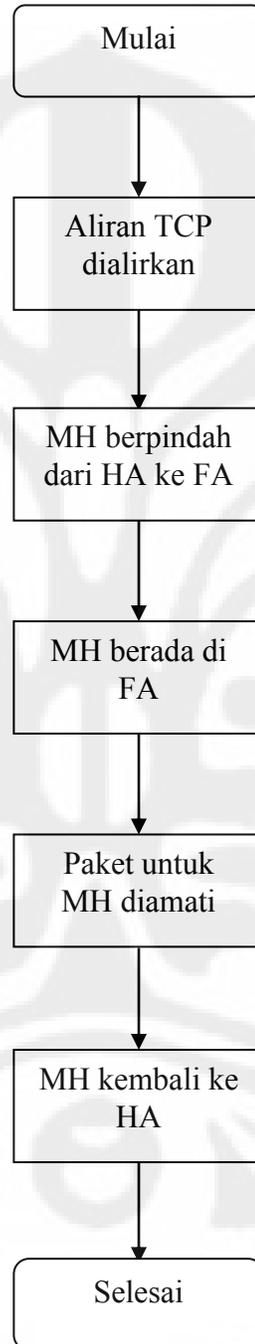
Dengan demikian, terdapat tiga buah domain pada topologi ini, yaitu satu domain *wired* (dilambangkan oleh 0) dan dua domain *wireless* (dilambangkan oleh 1 dan 2). Kemudian masing-masing *node* dari tiap domain diberi *address* secara hirarki. Dua *wired node* ditempatkan pada dua sub-domain (disebut dengan *cluster*) yang terpisah, yaitu 0 dan 1 sehingga *address* mereka adalah 0 (domain 0).0 (*cluster* 0).0 (hanya ada satu *node* pada *cluster* 0) dan 0 (domain 0).1 (*cluster* 1).0 (hanya ada satu *node* pada *cluster* 1). Pada domain *wireless* yang pertama (domain 1), *home agent* dan MH terletak pada satu *cluster* tunggal yang sama sehingga *address* mereka masing-masing adalah 1.0.0 dan 1.0.1. Sedangkan pada domain *wireless* yang kedua (domain 2) terdapat *foreign agent* dengan *address* 2.0.0.

Untuk koneksi antara *wired* dan *base station node*, digunakan *duplex link* dengan *bandwidth* sebesar 5Mb, *delay* 2 ms, dan antrean *DropTail*. Sedangkan antara *base station node* dan MH digunakan *bandwidth* sebesar 10 Mb dan *delay* 10 ms. Untuk memancarkan sinyal nirkabel, pada *base station node* digunakan antena jenis *omnidirectional*. Teknologi nirkabel yang digunakan adalah *Wireless Local Area Network* (WLAN) sebab modul WiMAX untuk NS belum tersedia. Keseluruhan *node* menggunakan tipe data *Transmission Control Protocol* (TCP). Kemudian, kedua *base station node* dikonfigurasi untuk menyalakan MIP *flag* dengan menggunakan opsi “–MobileIP ON”. Setelah itu, MH dikonfigurasi sehingga berpindah ke domain *foreign agent* dengan kecepatan 20 m/s pada detik ke 100 dan kembali ke domain *home agent* pada detik ke 200. Demikian juga halnya pada variasi kecepatan 33 m/s.

### 3.4 Alur Simulasi

Simulasi diawali dengan dialirkannya aliran TCP antara W0 dan MH. Kemudian, MH akan berpindah dari *home agent* ke *foreign agent* pada detik ke 100 dan kembali ke *home agent* pada detik ke 200 dengan kecepatan 20 dan 33 m/s. Dengan menggunakan fitur grafik yang terdapat pada NAM, paket yang ditujukan untuk MH akan diamati. Keseluruhan simulasi berlangsung selama kurang lebih 250

detik. Secara keseluruhan, alur simulasi ditunjukkan pada gambar diagram alir berikut.



Gambar 3.5 Diagram alir simulasi

### 3.5 Parameter Simulasi

Parameter yang menjadi fokus pada simulasi ini adalah sebagai berikut:

1. *Bandwidth*

Pada simulasi ini akan digunakan fitur *graph bandwidth* yang terdapat pada NAM. *Graph bandwidth* akan merepresentasikan *traffic bandwidth* yang terjadi antara tiap-tiap *node*.

2. *Handover*

Ketika MH sedang berpindah dari HA ke FA, akan diamati paket-paket yang ditujukan untuk MH pada periode tersebut.

3. *Packet Delay*

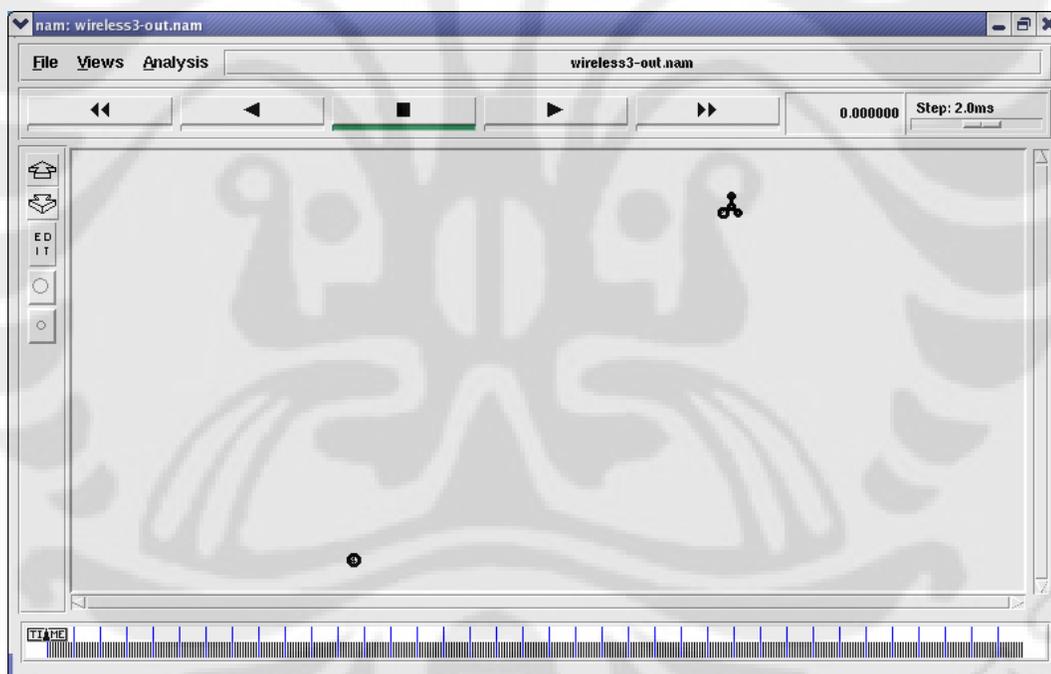
Waktu yang dibutuhkan oleh paket untuk dapat kembali terkirim pada saat MH telah melakukan *handover*.

Untuk mengamati parameter-parameter tersebut, digunakan fitur *graph* yang sudah terintegrasi pada NAM. Fitur *graph* diaktifkan dengan cara mengklik *link* di antara tiap-tiap *node*.

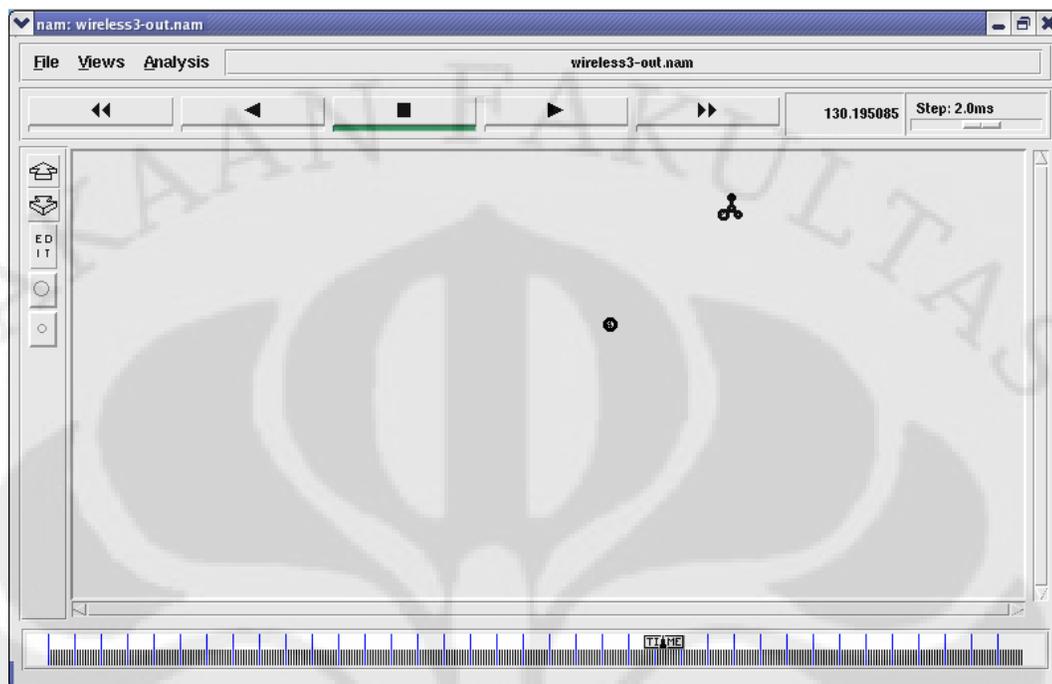


*Window* tersebut merupakan *window* pengendali simulasi. *Window* tersebut nantinya akan menganimasikan topologi jaringan yang telah dijelaskan pada bab sebelumnya. Pada *window* tersebut terdapat lima buah *node*, yaitu *node* 0, *node* 1, *node* 2, dan *node* 3 yang saling terhubung serta *node* 4 yang terletak di pojok kiri bawah, yang kesemuanya mewakili topologi jaringan yang telah dirancang. Dengan menekan tombol *play forward*, maka simulasi akan berjalan.

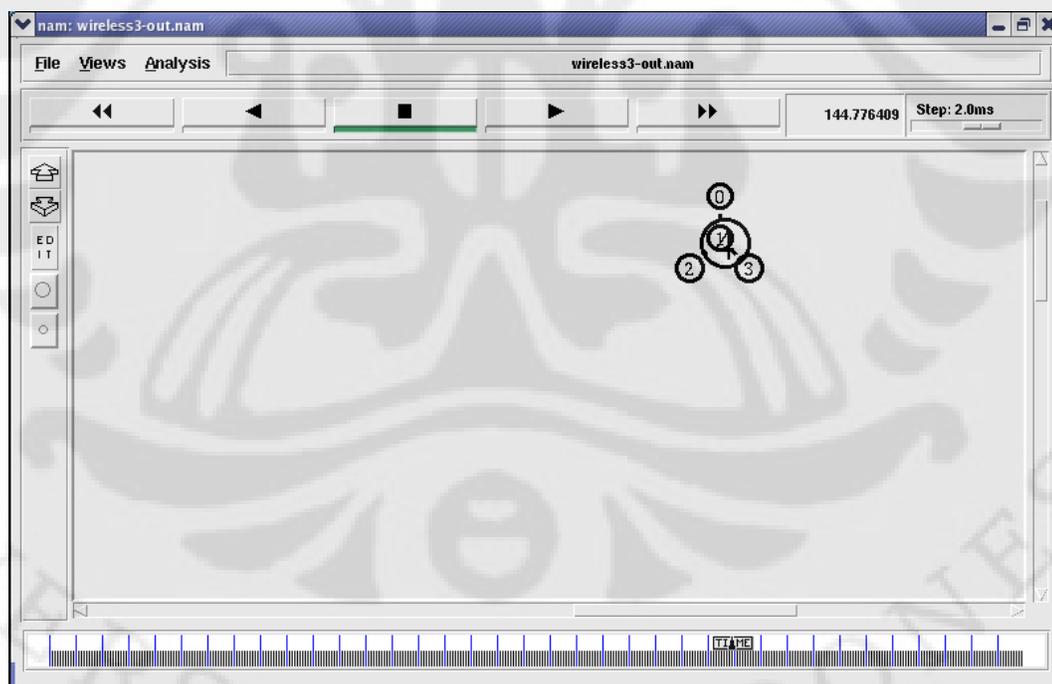
Setelah waktu simulasi telah berjalan selama 100 detik, MH mulai berpindah dari HA menuju FA seperti yang ditunjukkan pada gambar 4.3 hingga akhirnya tiba di FA seperti yang ditunjukkan pada gambar 4.4. Sedangkan gambar 4.5 menunjukkan MH kembali ke HA pada detik ke 200. Pada detik ke 250 simulasi berakhir seperti ditunjukkan pada gambar 4.6.



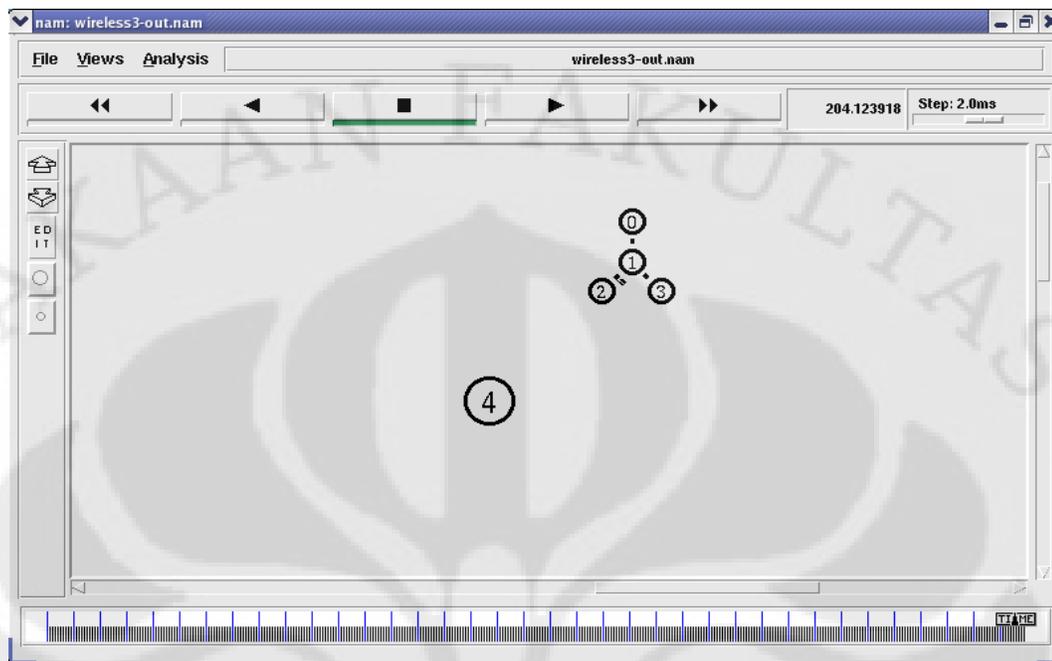
Gambar 4.2 *Window* simulasi



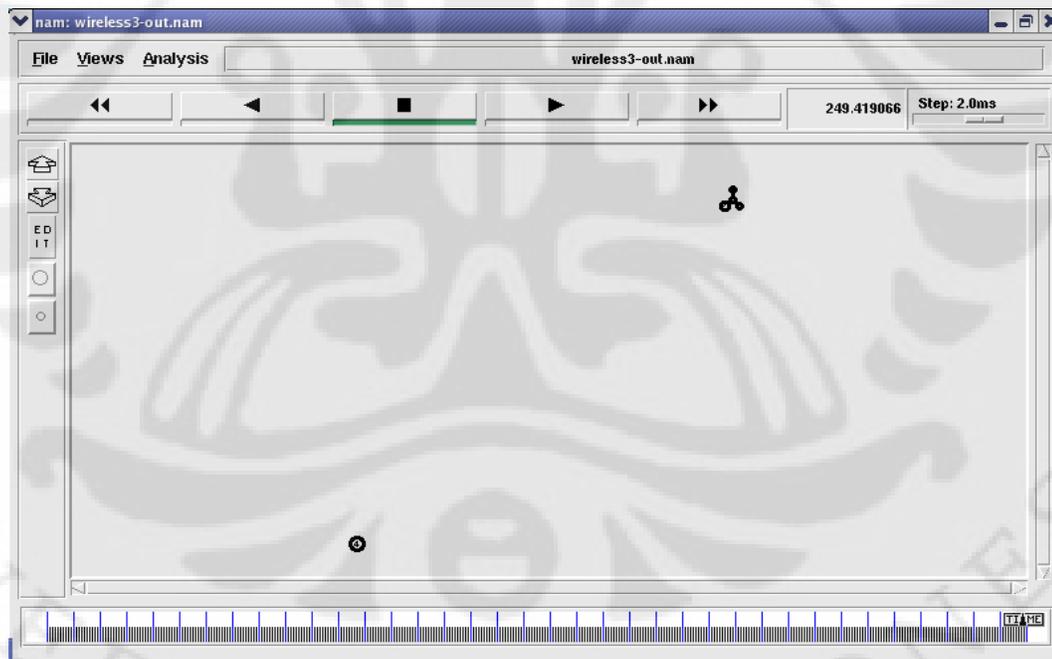
Gambar 4.3 MH berpindah dari HA ke FA



Gambar 4.4 MH tiba di FA



Gambar 4.5 MH kembali ke HA



Gambar 4.6 Akhir simulasi

## 4.2 Pengamatan Paket

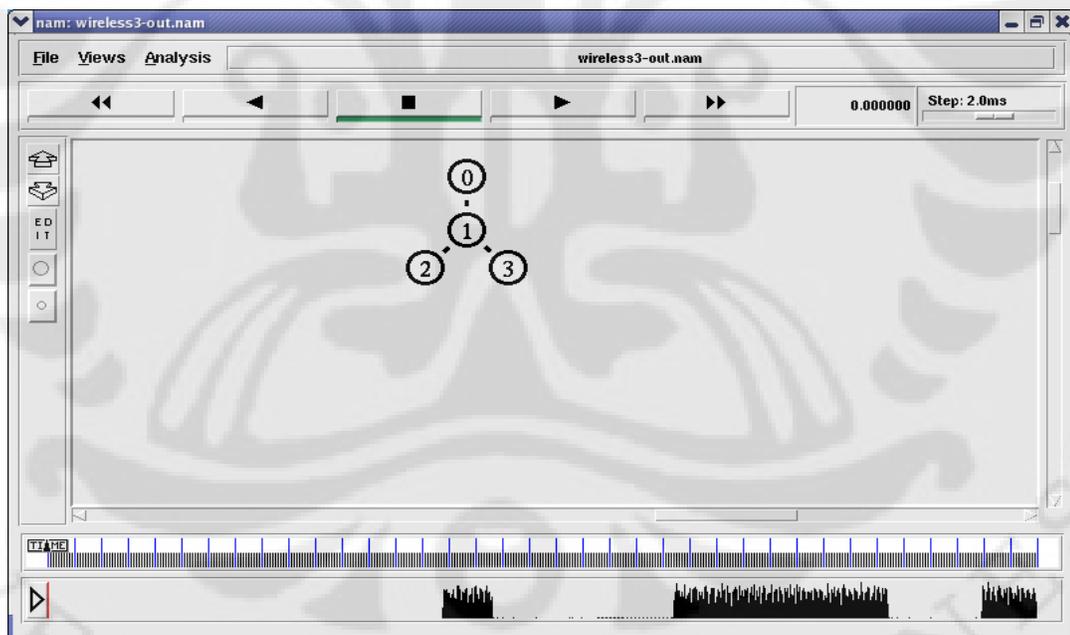
Dengan menggunakan fitur *graph bandwidth*, paket yang mengalir akan diamati. Paket yang akan diamati difokuskan pada *link* antara *node* 1 dan *node* 2, *link* antara *node* 1 dan *node* 3, dan *link* antara *node* 2 dan *node* 3.

### 4.2.1 Graph Bandwidth

*Graph bandwidth* akan merepresentasikan *traffic bandwidth* yang terjadi antara tiap-tiap *node*. Grafik tersebut berupa pulsa, dimana sumbu-x merepresentasikan waktu dalam detik dan sumbu-y merepresentasikan level sinyal *bandwidth*. Perbesaran gambar grafik terdapat pada lampiran 2 dan 3.

#### 4.2.1.1 Bandwidth Node 1 – Node 2

Berikut ini adalah grafik *bandwidth* antara *node* 1 dan *node* 2 (HA).

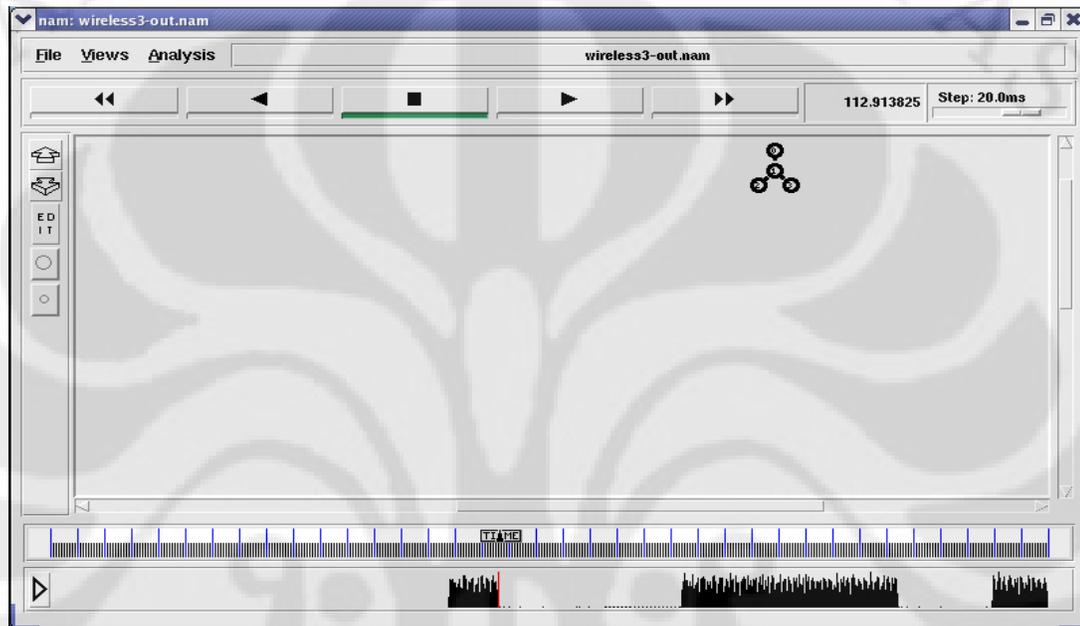


Gambar 4.7 Grafik *bandwidth* antara *node* 1 dan *node* 2

Grafik tersebut menunjukkan paket yang dikirimkan dari *node* 1 ke HA dan diteruskan ke MH sehingga keseluruhan aliran paket yang ditujukan untuk MH dapat

**Universitas Indonesia**

terlihat melalui grafik ini atau dengan kata lain, semua paket yang dikirimkan oleh *node* 1 ditujukan untuk MH, baik melalui *node* 2 maupun *node* 3. Paket mulai dikirimkan pada detik ke 100, yaitu ketika MH mulai bergerak untuk berpindah ke FA.



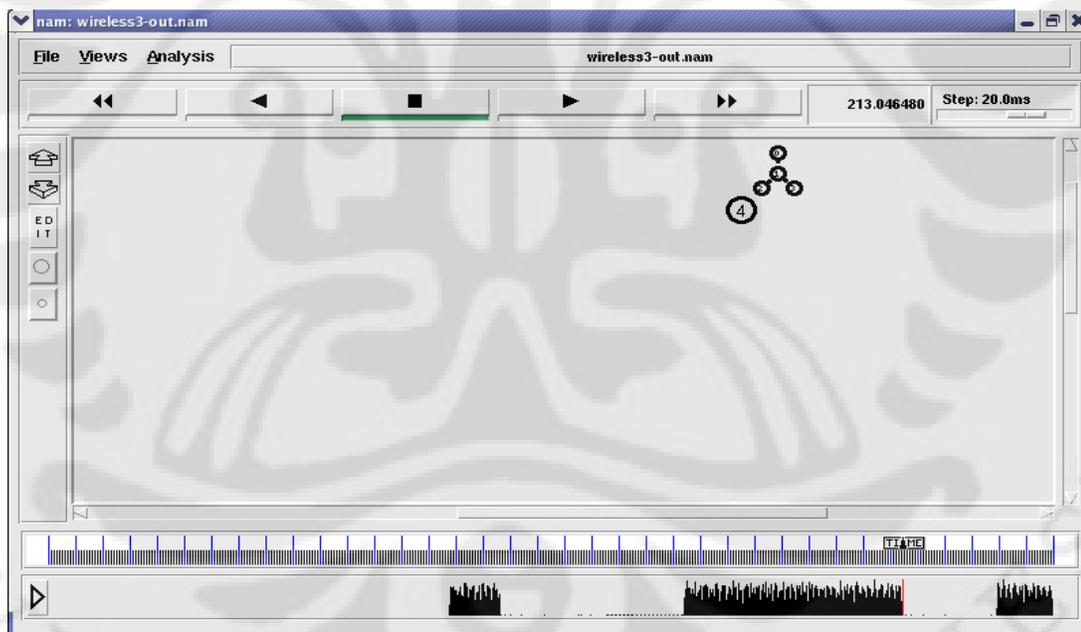
Gambar 4.8 MH mulai meninggalkan HA

Pada detik ke 113, MH mulai meninggalkan HA, terlihat dari grafik yang mengecil dan terputus untuk beberapa detik seperti yang terlihat pada gambar 4.8 di atas. Pada saat tersebut terjadi *handover*. Pada detik ke 138, grafik mulai menyambung kembali dan benar-benar tersambung pada detik ke 158 yang menunjukkan MH sudah tiba di FA. MH mulai benar-benar meninggalkan FA pada detik ke 213 dan tiba kembali di HA pada detik ke 236. Waktu yang dibutuhkan untuk berpindah dari HA ke FA dan sebaliknya masing-masing kurang lebih sebesar 45 detik dan 23 detik sehingga rata-rata *delay* yang dibutuhkan untuk satu kali *handover* pada kecepatan 20 m/s adalah 34 detik. Perbedaan *delay* tersebut disebabkan adanya *care of address acquisition* (CoA *acquisition*) ketika MH berpindah dari *home address*-nya ke jaringan yang lain.

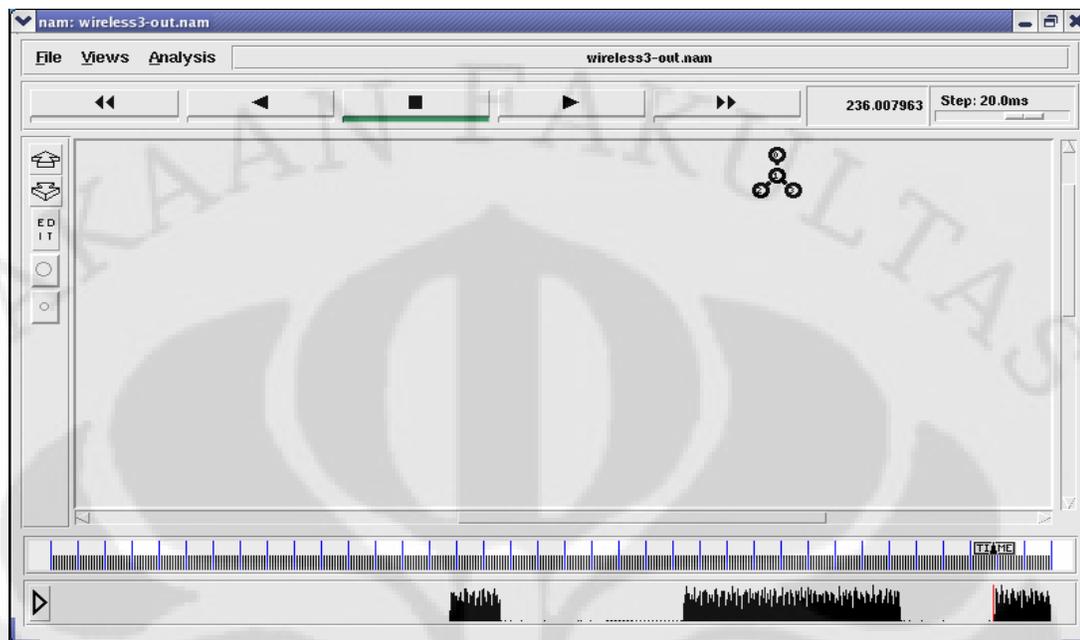
Universitas Indonesia



Gambar 4.9 MH tiba di FA



Gambar 4.10 MH meninggalkan FA



Gambar 4.11 MH tiba kembali di HA

#### 4.2.1.2 Bandwidth Node 1 – Node 3

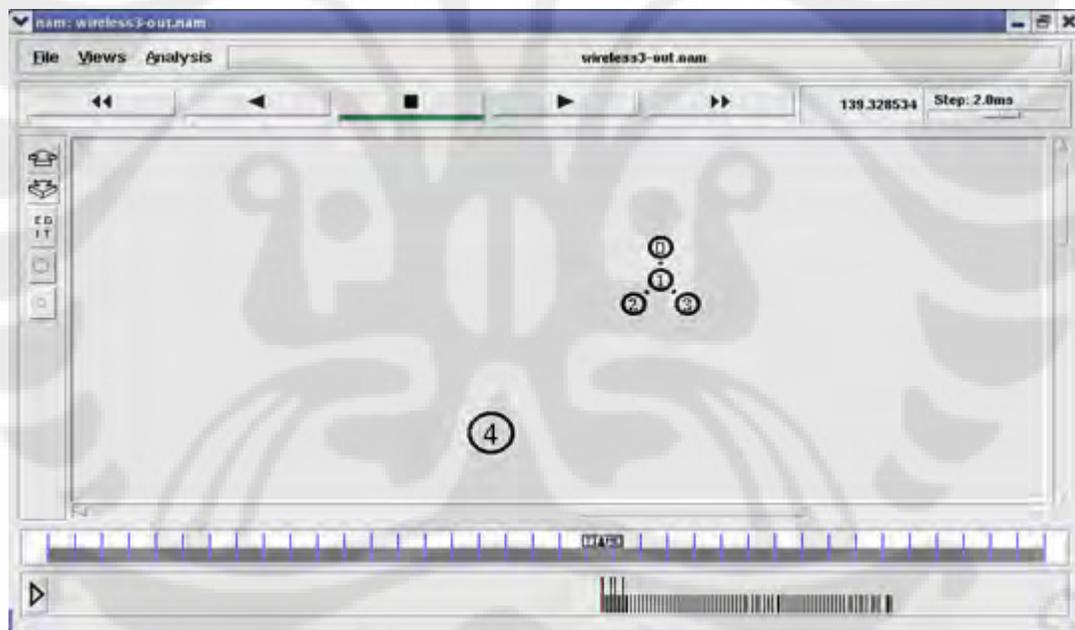
Berikut ini adalah grafik *bandwidth* antara *node 1* dan *node 3* (FA).

Gambar 4.12 Grafik *bandwidth* antara *node 1* dan *node 3*

Grafik tersebut menunjukkan paket yang dikirimkan dari *node* 1 ke FA. Dengan kata lain, *bandwidth* antara *node* 1 dan FA merepresentasikan *bandwidth* antara FA dan MH. Paket tersebut merupakan paket yang diteruskan oleh HA ketika MH sedang berada di FA. Terlihat pada gambar bahwa grafik mulai muncul pada detik ke 138 dan benar-benar muncul pada detik ke 158, sedangkan pada detik yang lain paket tidak dikirimkan melalui FA. Hal tersebut menunjukkan bahwa pada rentang waktu tersebut, MH sedang berada di FA. Hal ini sesuai dengan jalannya simulasi seperti yang telah dijelaskan sebelumnya.

#### 4.2.1.3 *Bandwidth Node 2 – Node 3*

Berikut ini adalah grafik *bandwidth* antara *node* 2 (HA) dan *node* 3 (FA).



Gambar 4.13 Grafik *bandwidth* antara *node* 2 dan *node* 3

Grafik antara HA dan FA ini hanya muncul ketika MH sedang berada di FA. Paket yang ditujukan untuk MH diteruskan melalui *node* 1 dan sebagian lagi diteruskan langsung ke FA oleh HA. Terlihat pada gambar bahwa interval grafik antara HA dan

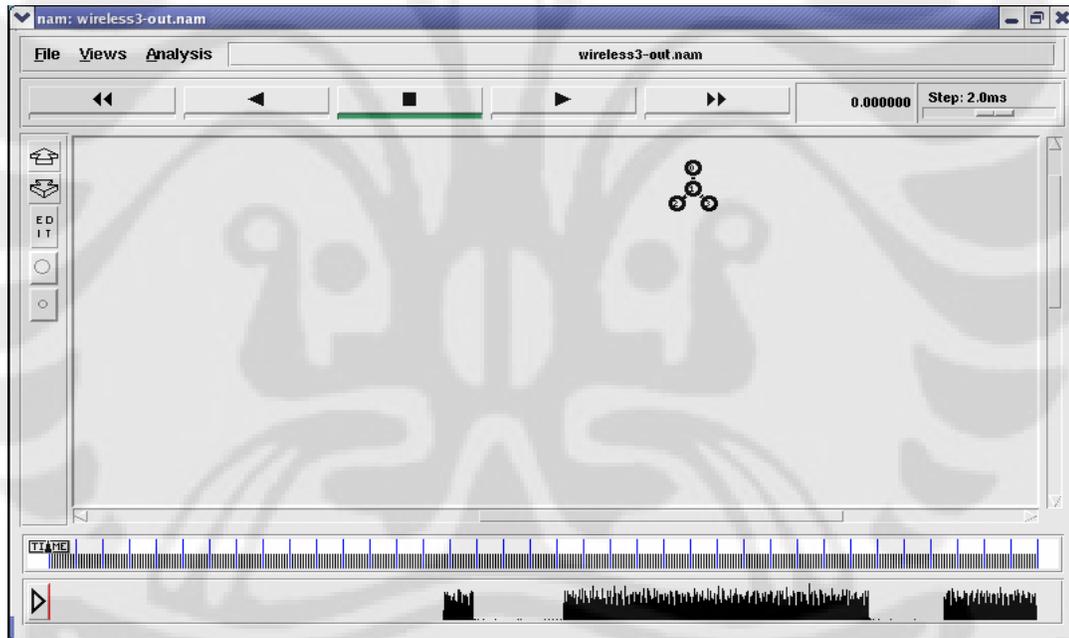
FA sama dengan interval grafik antara *node* 1 dan FA, namun grafiknya tipis. Hal ini dikarenakan sebagian besar paket diteruskan melalui *node* 1.

### 4.3 Variasi Kecepatan *Mobile Host* (MH)

Pada simulasi ini, keseluruhan skema dan alur simulasi tetap sama seperti pada simulasi sebelumnya. Hanya kecepatan MH diubah menjadi 33 m/s dari sebelumnya 20 m/s.

#### 4.3.1 *Bandwidth Node 1 – Node 2*

Berikut ini adalah grafik *bandwidth* antara *node* 1 dan *node* 2 (HA).

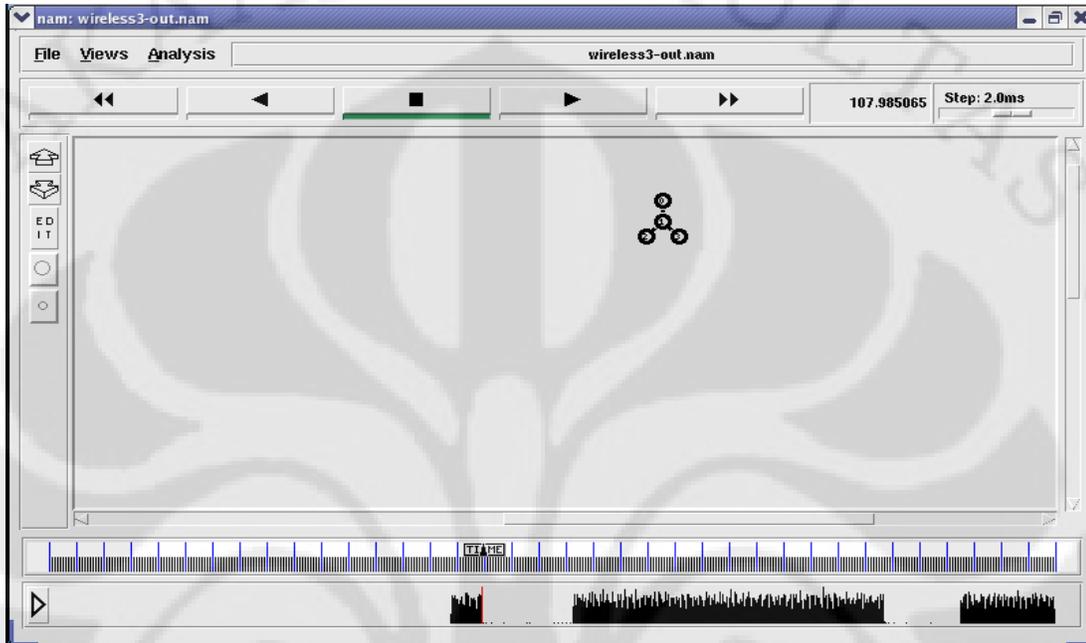


Gambar 4.14 Grafik *bandwidth* antara *node* 1 dan *node* 2

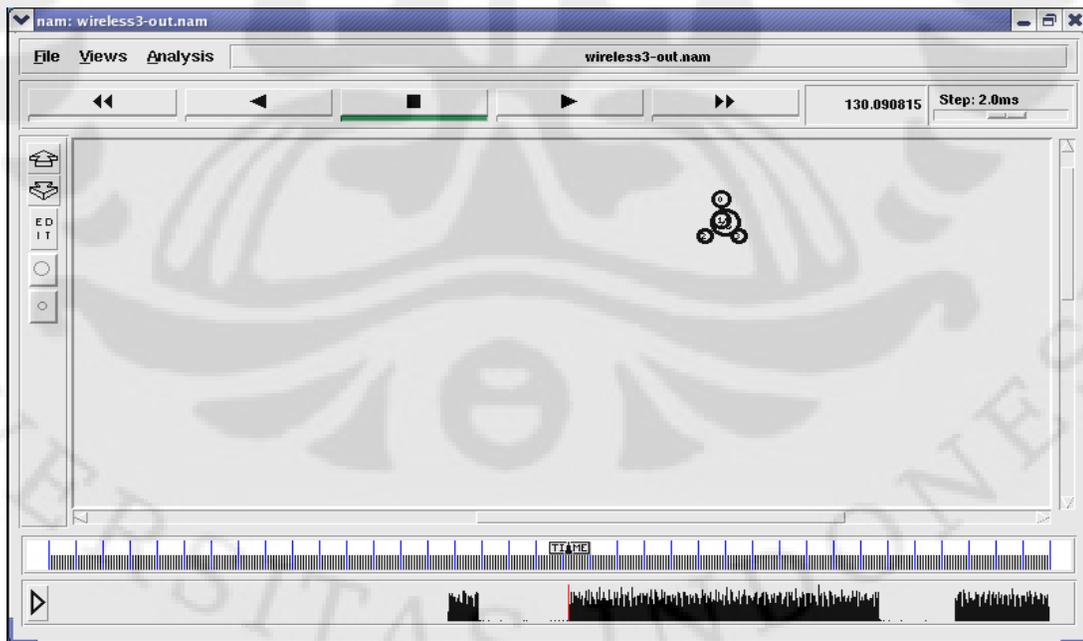
Berbeda dengan sebelumnya, pada kecepatan 33 m/s, *delay* yang dibutuhkan lebih pendek. Terlihat pada grafik bahwa MH mulai meninggalkan HA pada detik ke 108 dan tiba di FA pada detik ke 130. Kemudian MH meninggalkan FA pada detik ke 208 dan tiba kembali di HA pada detik 226. Waktu yang dibutuhkan untuk berpindah dari HA ke FA dan sebaliknya masing-masing kurang lebih sebesar 22 detik dan 18 detik

**Universitas Indonesia**

sehingga rata-rata *delay* yang dibutuhkan untuk satu kali *handover* pada kecepatan 33 m/s adalah 20 detik.

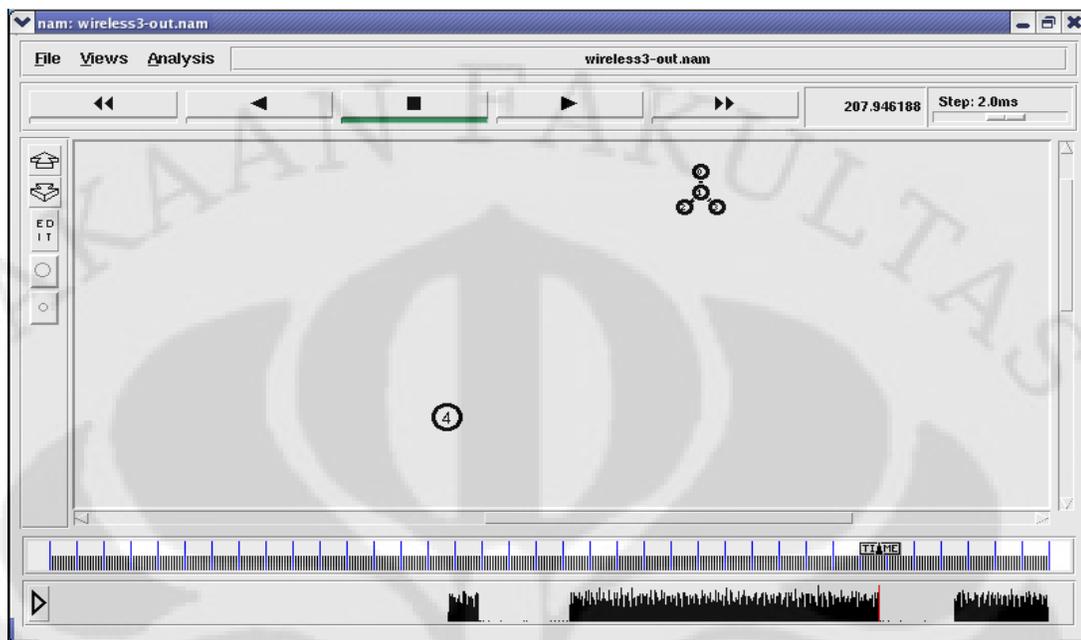


Gambar 4.15 MH mulai meninggalkan HA

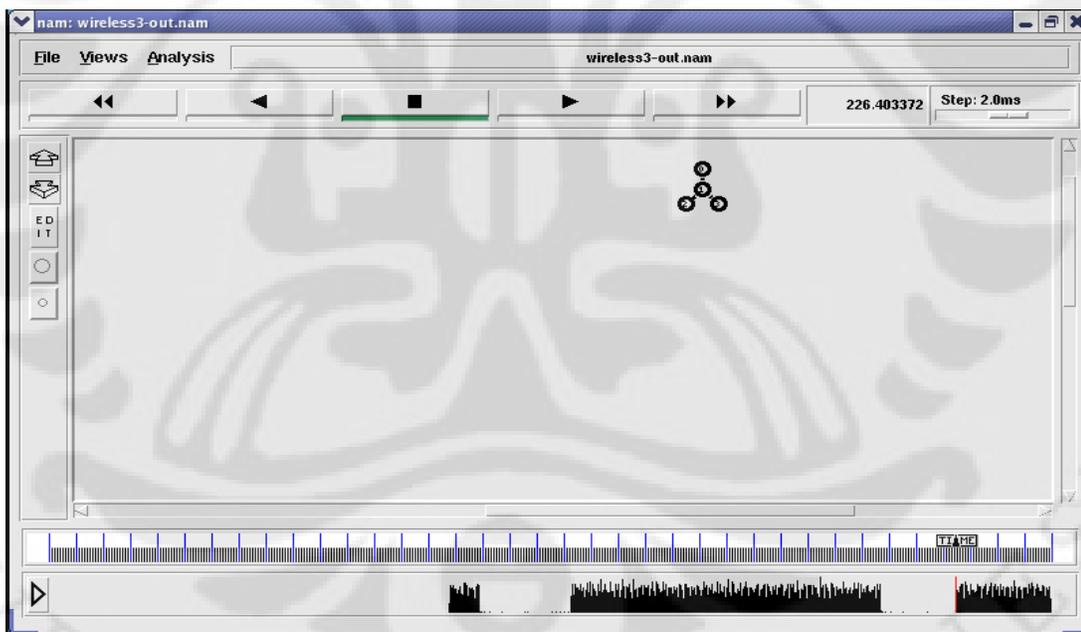


Gambar 4.16 MH tiba di FA

Universitas Indonesia



Gambar 4.17 MH meninggalkan FA



Gambar 4.18 MH tiba kembali di HA

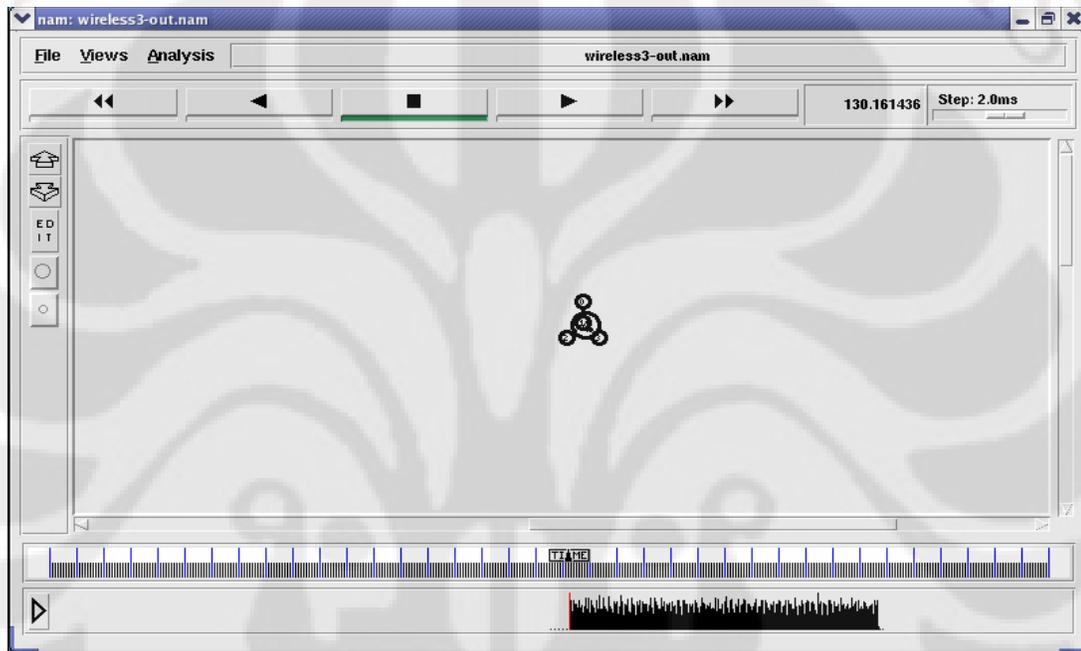
Pada gambar juga terlihat *range* grafik *bandwidth* yang dihasilkan lebih lebar jika dibandingkan dengan grafik yang sama pada kecepatan 20 m/s. Hal ini

**Universitas Indonesia**

merupakan akibat dari efek *Doppler* yang muncul akibat adanya penambahan kecepatan pada MH.

#### 4.3.2 *Bandwidth Node 1 – Node 3*

Berikut ini adalah grafik *bandwidth* antara *node 1* dan *node 3* (FA).



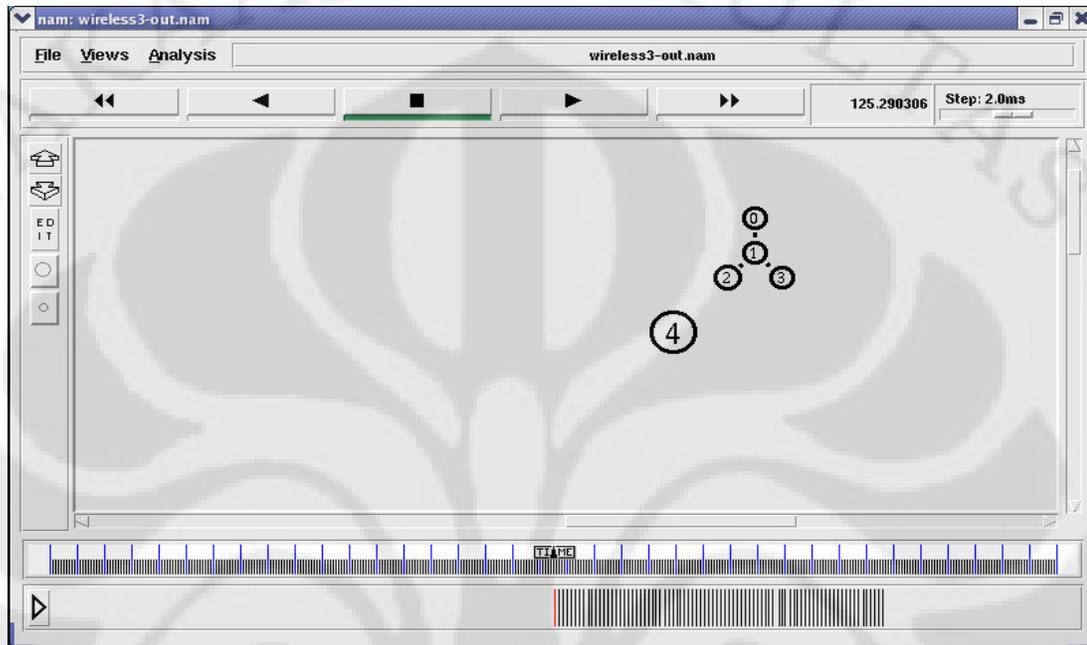
Gambar 4.19 Grafik *bandwidth* antara *node 1* dan *node 3*

Grafik ini menunjukkan paket yang diteruskan oleh HA ketika MH sedang berada di FA. Terlihat pada gambar bahwa grafik mulai muncul pada detik ke 126 dan benar-benar muncul pada detik ke 130, sedangkan pada detik yang lain paket tidak dikirimkan melalui FA.

Hal tersebut menunjukkan bahwa pada rentang waktu tersebut, MH sedang berada di FA. Perbedaan terlihat pada *range* grafik *bandwidth* yang dihasilkan, yaitu lebih lebar jika dibandingkan dengan grafik yang sama pada kecepatan 20 m/s. Hal ini menunjukkan bahwa simulasi telah berjalan sesuai dengan teori yang telah dijelaskan pada bab sebelumnya.

### 4.3.3 Bandwidth Node 2 – Node 3

Berikut ini adalah grafik *bandwidth* antara *node 2* (HA) dan *node 3* (FA).



Gambar 4.20 Grafik *bandwidth* antara *node 2* dan *node 3*

Sama seperti pada simulasi sebelumnya, grafik antara HA dan FA ini hanya muncul ketika MH sedang berada di FA. Paket yang ditujukan untuk MH diteruskan melalui *node 1* dan sebagian lagi diteruskan langsung ke FA oleh HA. Terlihat pada gambar bahwa interval grafik antara HA dan FA sama dengan interval grafik antara *node 1* dan FA, namun grafiknya tipis. Hal ini dikarenakan sebagian besar paket diteruskan melalui *node 1*.

## BAB V KESIMPULAN

Berdasarkan hasil simulasi dan analisis yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. *Mobile IP (MIP)* memungkinkan *Mobile Host (MH)* untuk dapat tetap menerima paket data meskipun MH sedang berada di FA.
2. *Delay* yang dibutuhkan oleh MH untuk berpindah dari HA ke FA dan dari FA ke HA pada kecepatan 20 m/s adalah 45 detik dan 23 detik sedangkan pada kecepatan 33 m/s adalah 22 detik dan 18 detik.
3. Perbedaan *delay* ketika MH berpindah dari HA ke FA dan dari FA ke HA disebabkan adanya *care of address acquisition (CoA acquisition)* ketika MH berpindah dari *home address*-nya ke jaringan yang lain.
4. Rata-rata *delay* yang dibutuhkan untuk satu kali *handover* oleh suatu MH pada kecepatan 20 m/s dengan menggunakan MIP adalah 34 detik dan pada kecepatan 33 m/s adalah 20 detik.
5. *Delay* yang cukup lama tersebut disebabkan oleh penggunaan standar nirkabel 802.11 pada simulasi dan efek *Doppler*, yaitu pengaruh kecepatan dari MH relatif terhadap *base station*. Semakin cepat kecepatan MH, maka *delay* yang dibutuhkan lebih pendek.
6. Dengan menggunakan MIP, akses internet di kereta api memungkinkan untuk dilakukan.

**DAFTAR ACUAN**

- [1] *Handover Komunikasi Seluler*. Diakses 13 Maret 2009  
<http://telekomui.org/?p=11>
- [2] *General Packet Radio Services*. Diakses 13 Maret 2009  
<http://id.wikipedia.org/wiki/GPRS>
- [3] *Enhanced Data rates for GSM Evolution*. Diakses 13 Maret 2009  
[http://id.wikipedia.org/wiki/Enhanced\\_Data\\_Rates\\_for\\_GSM\\_Evolution](http://id.wikipedia.org/wiki/Enhanced_Data_Rates_for_GSM_Evolution)
- [4] *Universal Mobile Telecommunication System*. Diakses 13 Maret 2009  
[http://www.ittelkom.ac.id/library/index.php?view=article&catid=17%3Asistem-komunikasi-bergerak&id=75%3Aumts-universal-mobile-telecommunication-system-&option=com\\_content&Itemid=15](http://www.ittelkom.ac.id/library/index.php?view=article&catid=17%3Asistem-komunikasi-bergerak&id=75%3Aumts-universal-mobile-telecommunication-system-&option=com_content&Itemid=15)
- [5] *High-Speed Downlink Packet Access*. Diakses 13 Maret 2009  
[http://id.wikipedia.org/wiki/High-Speed\\_Downlink\\_Packet\\_Access](http://id.wikipedia.org/wiki/High-Speed_Downlink_Packet_Access)
- [6] *Quality of Service*. Diakses 14 Maret 2009  
<http://telkom.tk/wiki/QoS>
- [7] Pareit et al. 2007. *QoS-enabled Internet-on-train network architecture: interworking by MMP-SCTP versus MIP*. Diakses 25 Februari 2009  
<http://www.pats.ua.ac.be/content/publications/2007/sctpvsmip.pdf>
- [8] *Mobile IP Overview, History, and Motivation*. Diakses 12 Juni 2009  
[http://www.tcpipguide.com/free/t\\_MobileIPOverviewHistoryandMotivation-3.htm](http://www.tcpipguide.com/free/t_MobileIPOverviewHistoryandMotivation-3.htm)
- [9] Chung, J., Claypool, M. June 2002. *NS by Example*. Diakses 12 Juni 2009  
<http://nile.wpi.edu/NS/>

## DAFTAR PUSTAKA

Pareit et al, 2007, *QoS-enabled Internet-on-train network architecture: inter-working by MMP-SCTP versus MIP*.

Diakses 25 Februari 2009.

<http://www.pats.ua.ac.be/content/publications/2007/sctpvsmip.pdf>

IBBT-UGent-IBCN, IBBT-UAntwerpen-PATS. 2007. *Internet on Train Architecture for service continuity*.

Diakses 26 Februari 2009.

<http://ew.thales.no/proceedings/2.4%20Internet%20on%20Train%20-%20Architecture%20for%20Service%20Continuity.pdf>

Pedoman Penulisan Tugas Akhir, Universitas Indonesia, Depok, 2008.

## LAMPIRAN

### Lampiran 1. Listing program

```
# wireless3.tcl
# simulation of a wired-cum-wireless topology running with mobileIP
#
=====
==
# 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) 1 ;# number of
mobilenodes
set opt(adhocRouting) DSDV ;# routing protocol

set opt(cp) "" ;# cp file not used
set opt(sc) "" ;# node movement
file.

set opt(x) 670 ;# x coordinate of
topology
set opt(y) 670 ;# y coordinate of
topology
set opt(seed) 0.0 ;# random seed
set opt(stop) 250 ;# time to stop
simulation

set opt(ftp1-start) 100.0

set num_wired_nodes 2
#set num_bs_nodes 2 ; this is not really used here.
```

```

#
=====
==

# check for boundary parameters and random seed
if { $opt(x) == 0 || $opt(y) == 0 } {
    puts "No X-Y boundary values given for wireless topology\n"
}
if {$opt(seed) > 0} {
    puts "Seeding Random number generator with $opt(seed)\n"
    ns-random $opt(seed)
}

# create simulator instance
set ns_ [new Simulator]

# set up for hierarchical routing
$ns_ node-config -addressType hierarchical

AddrParams set domain_num_ 3                ;# number of domains
lappend cluster_num 2 1 1                    ;# number of clusters in each
domain
AddrParams set cluster_num_ $cluster_num
lappend eilastlevel 1 1 2 1                  ;# number of nodes in each
cluster
AddrParams set nodes_num_ $eilastlevel ;# of each domain

set tracefd [open wireless3-out.tr w]
set namtrace [open wireless3-out.nam w]
$ns_ trace-all $tracefd
$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)

# Create topography object
set topo [new Topography]

# define topology
$topo load_flatgrid $opt(x) $opt(y)

# create God
# 2 for HA and FA
create-god [expr $opt(nn) + 2]

# create wired nodes
set temp {0.0.0 0.1.0} ;# hierarchical addresses
for {set i 0} {$i < $num_wired_nodes} {incr i} {
    set W($i) [$ns_ node [lindex $temp $i]]
}

# Configure for ForeignAgent and HomeAgent nodes
$ns_ node-config -mobileIP ON \
                -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 ON \
        -agentTrace ON \
        -routerTrace OFF \
        -macTrace OFF

# Create HA and FA
set HA [$ns_ node 1.0.0]
set FA [$ns_ node 2.0.0]
$HA random-motion 0
$FA random-motion 0

# Position (fixed) for base-station nodes (HA & FA).
$HA set X_ 1.000000000000
$HA set Y_ 2.000000000000
$HA set Z_ 0.000000000000

$FA set X_ 650.000000000000
$FA set Y_ 600.000000000000
$FA set Z_ 0.000000000000

# create a mobilenode that would be moving between HA and FA.
# note address of MH indicates its in the same domain as HA.
$ns_ node-config -wiredRouting OFF

set MH [$ns_ node 1.0.1]
set node_(0) $MH
set HAaddress [AddrParams addr2id [$HA node-addr]]
[$MH set regagent_] set home_agent_ $HAaddress

# movement of the MH
$MH set Z_ 0.000000000000
$MH set Y_ 2.000000000000
$MH set X_ 2.000000000000

# MH starts to move towards FA
$ns_ at 100.000000000000 "$MH setdest 640.000000000000
610.000000000000 20/33.000000000000"
# goes back to HA
$ns_ at 200.000000000000 "$MH setdest 2.000000000000 2.000000000000
20/33.000000000000"

# create links between wired and BaseStation nodes
$ns_ duplex-link $W(0) $W(1) 5Mb 2ms DropTail
$ns_ duplex-link $W(1) $HA 5Mb 2ms DropTail
$ns_ duplex-link $W(1) $FA 5Mb 2ms DropTail

```

```

$ns_ duplex-link-op $W(0) $W(1) orient down
$ns_ duplex-link-op $W(1) $HA orient left-down
$ns_ duplex-link-op $W(1) $FA orient right-down

# setup TCP connections between a wired node and the MobileHost

set tcp1 [new Agent/TCP]
$tcp1 set class_ 2
set sink1 [new Agent/TCPSink]
$ns_ attach-agent $W(0) $tcp1
$ns_ attach-agent $MH $sink1
$ns_ connect $tcp1 $sink1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns_ at $opt(ftp1-start) "$ftp1 start"

# source connection-pattern and node-movement scripts
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..."
}

# Define initial node position in nam
for {set i 0} {$i < $opt(nn)} {incr i} {

    # 20 defines the node size in nam, must adjust it according to
    your
    # scenario
    # The function must be called after mobility model is defined

    $ns_ initial_node_pos $node_($i) 20
}

# Tell all nodes when the siulation ends
for {set i 0} {$i < $opt(nn)} {incr i} {
    $ns_ at $opt(stop).0 "$node_($i) reset";
}
$ns_ at $opt(stop).0 "$HA reset";
$ns_ at $opt(stop).0 "$FA reset";

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

```

```
$ns_ at $opt(stop).0001 "stop"
proc stop {} {
    global ns_ tracefd namtrace
    close $tracefd
    close $namtrace
}

# some useful headers for tracefile
puts $tracefd "M 0.0 nn $opt(nn) x $opt(x) y $opt(y) rp \
    $opt(adhocRouting)"
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
```

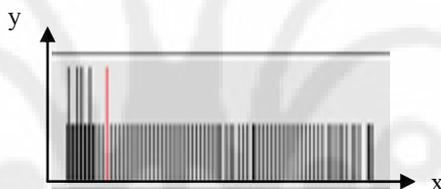
Lampiran 2. Perbesaran gambar grafik pada kecepatan 20 m/s



Grafik *bandwidth* antara node 1 dan node 2



Grafik *bandwidth* antara node 1 dan node 3



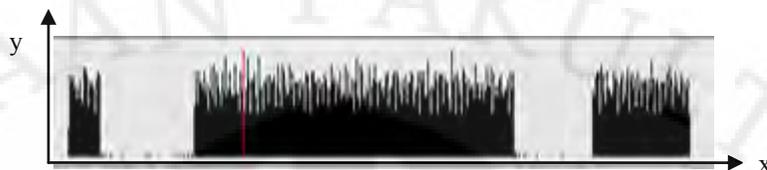
Grafik *bandwidth* antara node 2 dan node 3

Keterangan:

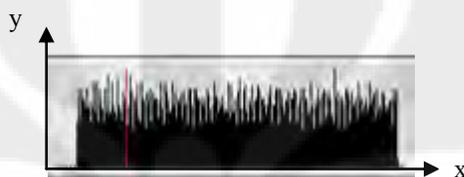
Sumbu-x = waktu (detik)

Sumbu-y = level sinyal *bandwidth*

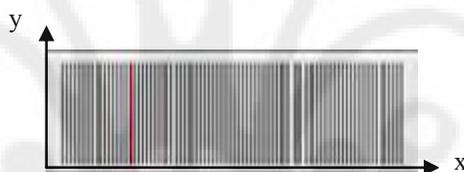
Lampiran 3. Perbesaran gambar grafik pada kecepatan 33 m/s



Grafik *bandwidth* antara *node 1* dan *node 2*



Grafik *bandwidth* antara *node 1* dan *node 3*



Grafik *bandwidth* antara *node 2* dan *node 3*

Keterangan:

Sumbu-x = waktu (detik)

Sumbu-y = level sinyal *bandwidth*