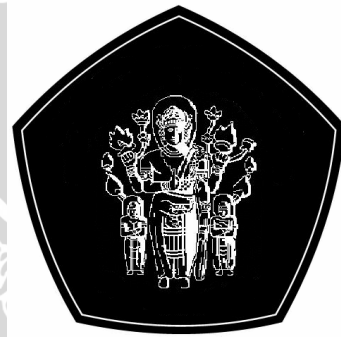


**ANALISIS PENERAPAN IPv6 UNTUK APLIKASI
VIDEOPHONE PADA JARINGAN UMTS**

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh :

YURI YUDO PRAMONO
NIM. 9901060419-63

DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
JURUSAN TEKNIK ELEKTRO
MALANG
2006

ANALISIS PENERAPAN IPv6 UNTUK APLIKASI VIDEOPHONE PADA JARINGAN UMTS

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

UNIVERSITAS BRAWIJAYA



Disusun oleh :

YURI YUDO PRAMONO
NIM. 9901060419-63

Telah diperiksa dan disetujui oleh
Dosen Pembimbing :

Ir. Endah Budi P, MT.
NIP. 131 839 362

Dwi Fadila K, ST. MT.
NIP. 132 258 185

ANALISIS PENERAPAN IPv6 UNTUK APLIKASI VIDEOPHONE PADA JARINGAN UMTS

Disusun oleh :

YURI YUDO PRAMONO
NIM. 9901060419-63

Skripsi ini telah diuji dan dinyatakan lulus pada
tanggal 27 Mei 2006

Majelis Penguji :

Ir. Wahyu Adi Priyono, MS.
NIP. 131 759 602

Ir. Erfan Achmad Dahlan
NIP. 131 124 663

Rudi Yuwono, ST, MSc.
NIP. 132 206 465

Gaguk Asmungi, ST.
NIP. 132 206 528

Mengetahui :
Ketua Jurusan Teknik Elektro

Ir. Purwanto, MT.
NIP. 131 574 847



FOR MY LIFE



KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Analisis Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS” ini. Penyusunan skripsi ini diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik di jurusan Teknik Elektro Universitas Brawijaya Malang.

Penulis menyadari selama penyusunan skripsi ini tidak terlepas dari bantuan, bimbingan, dorongan dan motivasi dari berbagai pihak. Oleh sebab itu, dengan kerendahan hati penulis menyampaikan terima kasih kepada :

- ❖ Ibu dan Bapak, serta kakak-kakakku untuk do'a dan dukungan yang telah diberikan selama studi hingga terselesaikannya skripsi ini.
- ❖ Bapak Ir. Purwanto, MT selaku Ketua Jurusan Teknik Elektro dan Bapak Ir. Heri Purnomo selaku Sekretaris Jurusan Teknik Elektro.
- ❖ Bapak Ir. Wahyu Adi P, MS selaku KKDK Teknik Telekomunikasi.
- ❖ Ibu Ir. Endah Budi P, MT dan Bapak Dwi Fadila K, ST. MT. selaku dosen pembimbing yang telah memberikan pengarahan dan bimbingan sampai terselesaikannya skripsi ini.
- ❖ Bapak dan Ibu dosen serta segenap staf dan karyawan Jurusan Teknik Elektro Universitas Brawijaya.
- ❖ Kawan-kawan Elektro 99, Mantan Dewan Teknik FT-UB 2001/2002 dan staf, Mantan Panitia KKM XXIV FT-UB 2002 (Sekum, Bendum, Kabid, Kasie and anggota, sepanjang sejarah teknik kayanya kita panitia yang paling kompak ya..), Shalahuddin Internet Crew dan kawan-kawan konsentrasi Teknik Telekomunikasi atas persahabatan, bantuan, masukan, semangat dan doanya.
- ❖ All keponakan yang lucu2, imut dan bandel2.

Penulis menyadari bahwa skripsi ini masih banyak kekurangan dan jauh dari kesempurnaan. Oleh karena itu kritik dan saran sangat diharapkan demi

kesempurnaan penyusunan skripsi ini. Akhir kata semoga skripsi ini dapat bermanfaat bagi rekan-rekan mahasiswa khususnya dan bagi seluruh pembaca pada umumnya.

Malang, Mei 2006

Penulis



RINGKASAN

YURI YUDO PRAMONO, Jurusan Teknik Elektro, Universitas Brawijaya Malang, Mei 2006, Analisis Penerapan IPv6 untuk Aplikasi *Vieophone* pada Jaringan UMTS, Dosen Pembimbing: Ir. Endah Budi P, MT dan Dwi Fadila K, ST.MT

IPv6 (*Internet Protocol version 6*) merupakan teknik pengalamatan baru yang memberikan jangkauan alamat yang lebih luas dibandingkan dengan IPv4. Aplikasi *videophone* merupakan aplikasi yang mengintegrasikan suara dan *video*. Memasuki milenium ketiga ini telah distandarkan sistem komunikasi bergerak generasi ketiga (3G) oleh *European Telecommunication Standar Institute* (ETSI) yang dikenal dengan *Universal Mobile Telecommunications System* (UMTS). Teknologi UMTS mampu memberikan kecepatan data sampai 2 Mbit/s serta menerapkan teknologi IP (*Internet Protocol*) di setiap *node* pada jaringan dan *user*nya.

Pada prinsipnya penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS adalah mengintegrasikan aplikasi *videophone* yang berbasis IPv6 dengan jaringan UMTS yang juga berbasis IPv6. Pada penulisan skripsi ini aplikasi *videophone* merujuk pada rekomendasi *International Telecommunication Union* (ITU-T) yaitu dengan menggunakan protokol H.323, dengan *CODEC video* yang digunakan adalah H.263 dan *CODEC audio* adalah G.723.1. Koneksi antara protokol H.323 dengan jaringan UMTS adalah dengan adanya *gatekeeper*. *Radius* yang digunakan adalah type sel *micro* dengan daerah operasi urban dan pusat kota yaitu sebesar 0,4 km dengan kecepatan data maksimal 384 Kbps.

Dalam analisis penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS diperoleh panjang paket aplikasi *videophone* terbesar yaitu 12189 bit atau 1523,625 byte dengan menggunakan *bandwith CODEC video* sebesar 384 Kbps, sedangkan panjang paket aplikasi *videophone* terkecil yaitu 2589 bit atau 323,625 byte dengan menggunakan *bandwith CODEC video* sebesar 64 Kbps. Kemudian diperoleh *delay end-to-end* terkecil yaitu 95,73445 ms dicapai pada saat menggunakan *bandwidth CODEC video* 64 kbps dengan panjang data 303,625 byte serta faktor utilisasi 0,1, sedangkan *delay end-to-end* terbesar yaitu 178,636391 ms dicapai pada saat menggunakan *bandwidth CODEC video* 384 kbps dengan panjang data 1503,625 byte serta faktor utilisasi 0,9. Probabilitas *packet loss total end-to-end* terbesar yaitu 0,7013 dengan menggunakan *bandwidth CODEC video* sebesar 384 kbps. Sedangkan probabilitas *packet loss total end-to-end* terkecil adalah 0,2313 dengan menggunakan *bandwidth CODEC video* 64 kbps. Serta diperoleh nilai *throughput* terbesar yaitu 248,0015125 bps, dicapai dengan menggunakan *segment* data sebesar 2589 bit dengan *bandwidth CODEC video* sebesar 64 kbps dan faktor utilisasi sebesar 0,1, sedangkan nilai *throughput* terkecil yaitu 86,07356342 bps dicapai dengan menggunakan *segment* data sebesar 12189 bit dengan *bandwidth CODEC video* sebesar 384 kbps dan faktor utilisasi sebesar 0,9.

DAFTAR ISI

HALAMAN JUDUL	
HALAMAN PENGESAHAN	
PENGANTAR	i
RINGKASAN	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	viii
DAFTAR TABEL	xi
DAFTAR SINGKATAN	xiii
DAFTAR SIMBOL	xv
BAB I PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	2
1.3. Ruang Lingkup	2
1.4. Tujuan.....	3
1.5. Sistematika Penulisan.....	3
BAB II INTERNET PROTOKOL VERSI 6 (IPv6) DAN VIDEOPHONE	
2.1 Internet Protokol Versi 6 (IPv6)	
2.1.1 Umum.....	4
2.1.2 Konfigurasi Protokol TCP/IP.....	5
2.1.2.1 <i>Network Acces Layer</i>	5
2.1.2.2 <i>Internet Layer</i>	6
2.1.2.3 <i>Host to host Transport Layer</i>	6
2.1.2.4 <i>Application Layer</i>	8
2.1.3 Pemrosesan Aliran Data.....	8
2.1.4 Format <i>Header IPv6</i>	11
2.1.5 <i>Extension Header IPv6</i>	12
2.1.6 <i>Maximum Transmission Unit (MTU) IPv6 / ukuran paket IPv6.</i>	14
2.1.7 Proses Enkapsulasi dan Dekapsulasi.....	14



2.2 Videophone

2.2.1	Umum.....	15
2.2.2	Standar Protokol H.323	15
2.2.2.1	RTP (<i>Realtime Transport Protocol</i>)	16
2.2.2.2	UDP (<i>User Datagram Protocol</i>).....	18
2.2.3	Komponen H.323	19
2.2.3.1	Terminal.....	19
2.2.3.2	Gateway	20
2.2.3.3	Gatekeeper	21
2.2.3.4	Multipoint Control Unit (MCU)	21
2.2.4	Mekanisme H.323 Dalam Melakukan <i>Call Setup</i>	22
2.2.4.1	Gatekeeper Discovery Dan Registrasi	22
2.2.4.2	Proses Pembentukan Hubungan / <i>Call Setup</i>	22
2.2.5	CODEC (<i>COder/DECoder</i>).....	25
2.2.6	Format Gambar	26
2.2.7	Paket Data Aplikasi <i>Videophone</i>	27
2.2.8	<i>Delay End-to-end</i> Aplikasi <i>Videophone</i>	29
2.2.9	Probabilitas <i>Packet Loss</i> Aplikasi <i>Videophone</i>	30

**BAB III PENERAPAN IPv6 UNTUK APLIKASI VIDEOPHONE
PADA JARINGAN UMTS**

3.1.	Umum.....	32
3.2.	Konsep Dasar UMTS	33
3.3.	Spektrum Frekuensi.....	34
3.4.	Konfigurasi UMTS	36
3.4.1	<i>Core Network</i> (CN)	37
3.4.2	<i>UMTS Terrestrial Radio Access Network</i> (UTRAN).....	38
3.4.3	<i>User Equipment</i> (UE).....	38
3.5	Daerah Layanan dan Struktur Sel UMTS.....	39
3.6	Jaringan Paket Data UMTS	42
3.7	Konfigurasi Aplikasi <i>Videophone</i> pada Jaringan UMTS	42
3.8	Mekanisme Pengiriman Paket Data <i>Videophone</i>	43
3.9	Konfigurasi <i>Protocol Stack</i> Jaringan UMTS berbasis IPv6.....	44

3.10	Parameter-parameter yang dianalisis.....	45
3.10.1	<i>Delay End-to-end</i>	45
3.10.1.1	<i>Delay</i> Proses	46
3.10.1.2	<i>Delay</i> Transmisi	58
3.10.1.3	<i>Delay</i> Propagasi	61
3.10.1.4	<i>Delay</i> Antrian.....	64
3.10.2	Probabilitas <i>Packet Loss</i> Jaringan UMTS.....	67
3.10.3	<i>Throughput</i>	70

BAB IV METODOLOGI PENELITIAN

4.1	Metode Pengambilan Data	73
4.1.1	Spesifikasi Internet Protokol versi 6 (IPv6).....	73
4.1.2	Spesifikasi Aplikasi <i>Videophone</i>	73
4.1.3	Spesifikasi Jaringan UMTS.....	74
4.2	Metode Pembatasan Variabel.....	74
4.3	Metode Analisis Data	75
4.4	Menarik Kesimpulan	75

BAB V ANALISIS PENERAPAN IPv6 UNTUK APLIKASI VIDEOPHONE PADA JARINGAN UMTS

5.1	Paket Data Aplikasi <i>Videophone</i>	76
5.2	<i>Delay End-to-end</i> Penerapan IPv6 untuk Aplikasi <i>Videophone</i> pada Jaringan UMTS	79
5.2.1	<i>Delay</i> Aplikasi <i>Videophone</i>	79
5.2.2	<i>Delay Gatekeeper</i>	80
5.2.3	<i>Delay</i> Jaringan UMTS.....	80
5.2.3.1	<i>Delay</i> Proses	81
5.2.3.2	<i>Delay</i> Transmisi	97
5.2.3.3	<i>Delay</i> Propagasi	101
5.2.3.4	<i>Delay</i> Antrian.....	105
5.3	Probabilitas <i>Packet Loss</i> Penerapan IPv6 untuk Aplikasi <i>Videophone</i> pada Jaringan UMTS.....	111
5.3.1	Probabilitas <i>Packet Loss</i> Aplikasi <i>Videophone</i>	111
5.3.2	Probabilitas <i>Packet Loss</i> Jaringan UMTS.....	112



5.4 *Throughput* 117

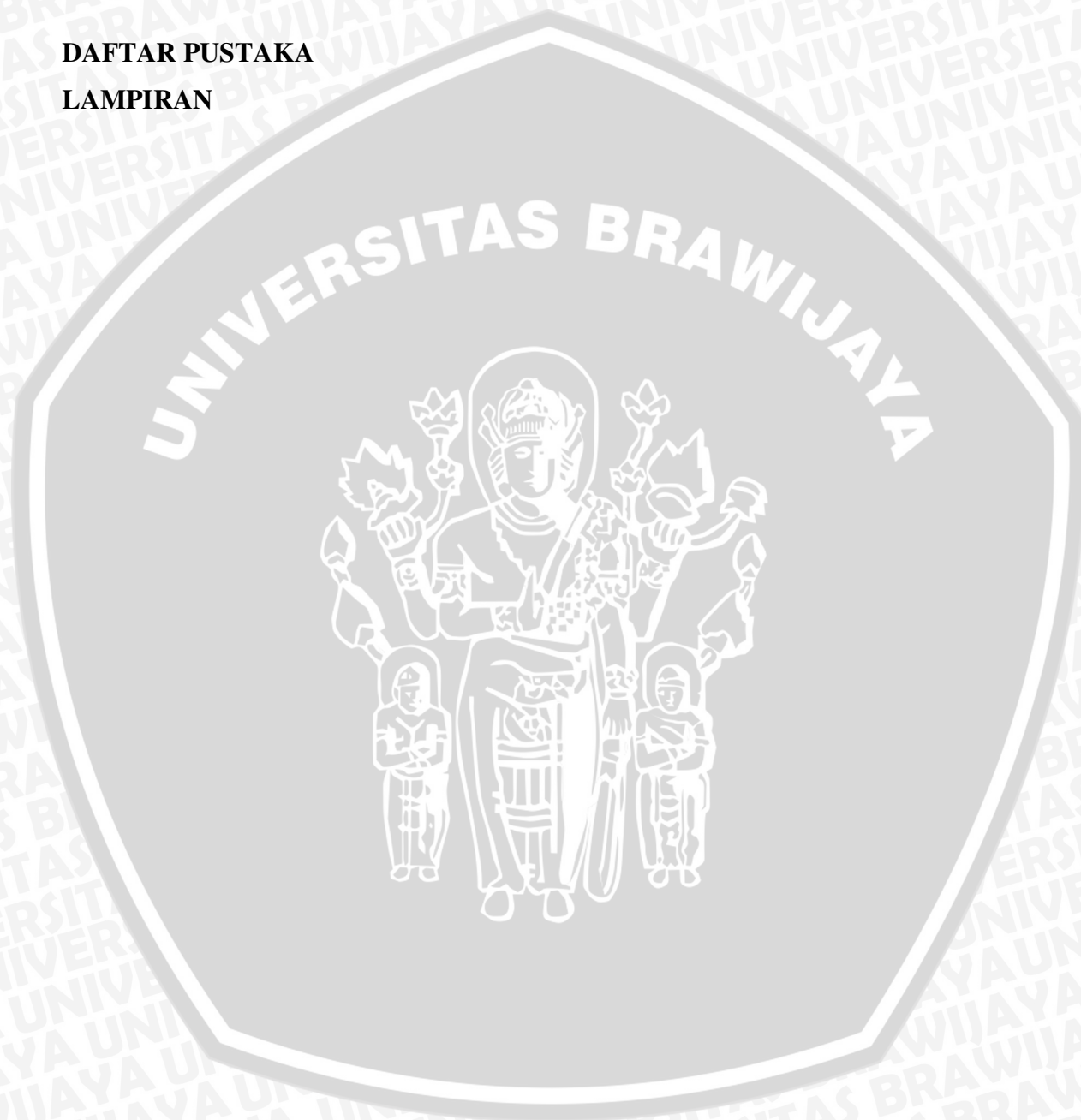
BAB VI PENUTUP

6.1. Kesimpulan..... 121

6.2. Saran 122

DAFTAR PUSTAKA

LAMPIRAN



DAFTAR GAMBAR

Gambar 2.1	Protokol pada tumpukan protokol TCP/IP	6
Gambar 2.2	Format segmen TCP	8
Gambar 2.3	Pemrosesan data selama pengiriman dan penerimaan	10
Gambar 2.4	Pergerakan data dalam <i>layer</i> TCP/IP	11
Gambar 2.5	Format <i>header</i> IPv6	11
Gambar 2.6	Format <i>extension header</i> pada paket IPv6	13
Gambar 2.7	Contoh pemakaian <i>extension header</i> pada IPv6	14
Gambar 2.8	Konfigurasi <i>videophone</i>	15
Gambar 2.9	H.323 <i>protocol stack</i>	16
Gambar 2.10	Format <i>header</i> RTP	16
Gambar 2.11	Format datagram UDP	18
Gambar 2.12	Komponen H.323	19
Gambar 2.13	H.323 <i>terminal equipment</i>	20
Gambar 2.14	Proses pembentukan hubungan aplikasi <i>videophone</i> menggunakan standar H.323 (menggunakan <i>direct endpoint call signaling</i>)	23
Gambar 2.15	Skema proses <i>call setup</i> antara 2 <i>endpoint</i> yang teregistrasi pada <i>gatekeeper</i> yang sama (menggunakan <i>direct endpoint call signalling</i>)	24
Gambar 2.16	Skema proses <i>call setup</i> antara 2 <i>endpoint</i> yang teregistrasi pada <i>gatekeeper</i> yang berbeda (menggunakan <i>direct endpoint call signaling</i>)	25
Gambar 2.17	Paket data <i>videophone</i>	28
Gambar 3.1	Skema perkembangan layanan sistem komunikasi seluler	33
Gambar 3.2	Konfigurasi Jaringan UMTS	37
Gambar 3.3	Zona global UMTS	39
Gambar 3.4	Konfigurasi jaringan paket data UMTS	42
Gambar 3.5	Konfigurasi aplikasi <i>videophone</i> pada jaringan UMTS	43
Gambar 3.6	Alur panggilan <i>videophone</i> pada jaringan UMTS	44
Gambar 3.7	Konfigurasi protokol IPv6 pada jaringan UMTS	44

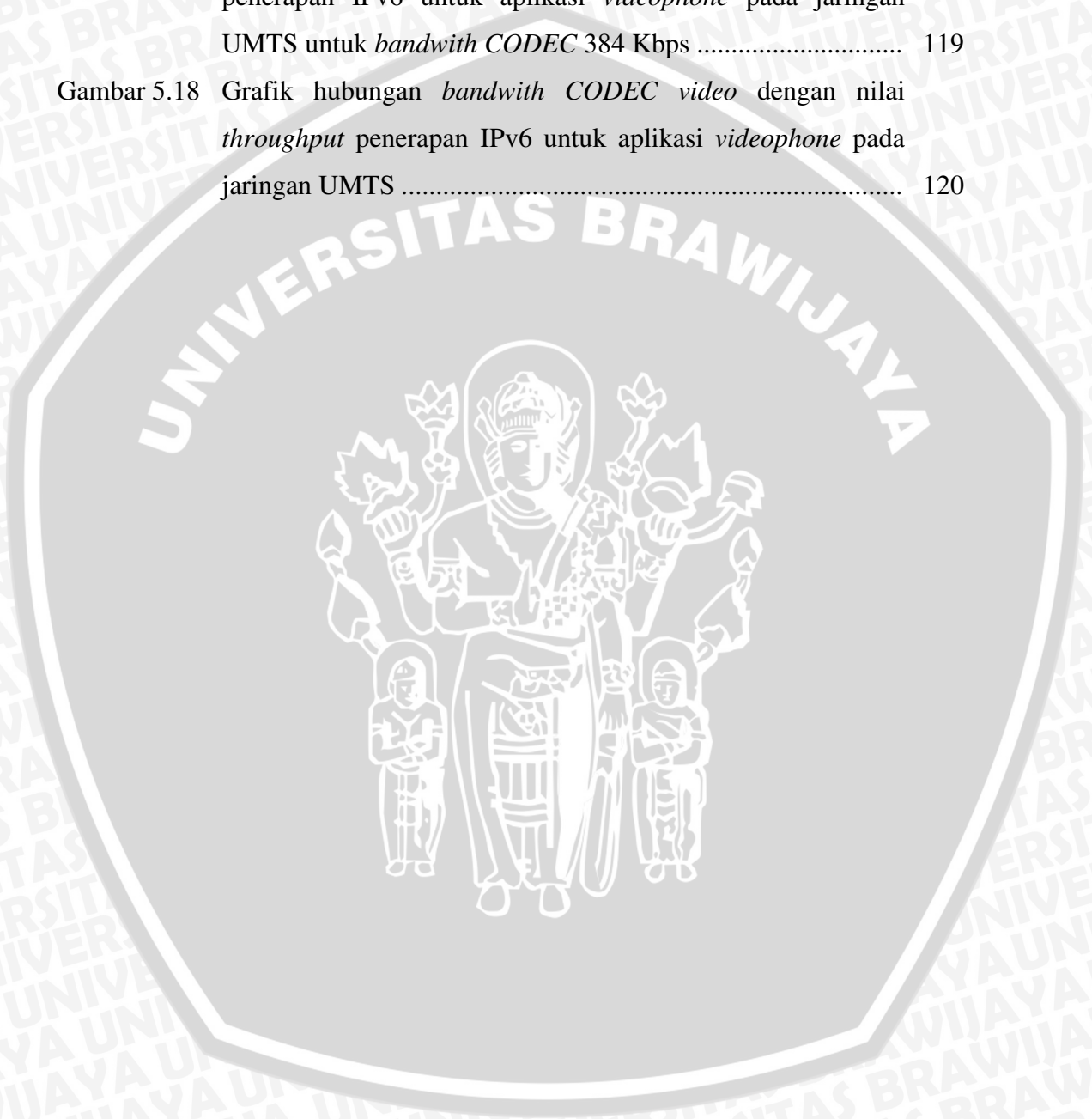
Gambar 3.8	Format PDCP data PDU	47
Gambar 3.9	Model antrian <i>single server</i>	66
Gambar 5.1	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya paket data (<i>payload</i>) aplikasi <i>videophone</i> berbasis IPv6	78
Gambar 5.2	Grafik hubungan <i>bandwith CODEC video</i> dengan <i>bandwith</i> aktual aplikasi <i>videophone</i> berbasis IPv6	78
Gambar 5.3	Analisis <i>delay end-to-end</i>	79
Gambar 5.4	Analisis <i>delay</i> proses pada jaringan UMTS.....	81
Gambar 5.5	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya <i>delay</i> proses pada jaringan UMTS.....	96
Gambar 5.6	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya <i>delay</i> proses pada jaringan UMTS	96
Gambar 5.7	Analisis <i>delay</i> transmisi pada jaringan UMTS	97
Gambar 5.8	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya <i>delay</i> transmisi pada jaringan UMTS.....	100
Gambar 5.9	Analisis <i>delay</i> propagasi pada jaringan UMTS.....	101
Gambar 5.10	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya <i>delay</i> propagasi pada jaringan UMTS.....	105
Gambar 5.11	Grafik hubungan faktor utilisasi dengan besarnya <i>delay</i> antrian pada jaringan UMTS.....	107
Gambar 5.12	Grafik hubungan faktor utilisasi dengan besarnya <i>delay</i> jaringan untuk <i>bandwith CODEC</i> 64 Kbps pada jaringan UMTS	109
Gambar 5.13	Grafik hubungan faktor utilisasi dengan besarnya <i>delay end-to-end</i> untuk <i>bandwith CODEC</i> 64 Kbps	110
Gambar 5.14	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya probabilitas <i>packet loss</i> total saat lintasan <i>forward</i> dan <i>reverse</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	116
Gambar 5.15	Grafik hubungan <i>bandwith CODEC video</i> dengan besarnya probabilitas <i>packet loss</i> total <i>end-to-end</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS.....	116



Gambar 5.16 Grafik hubungan faktor utilisasi dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *bandwith CODEC* 64 Kbps 119

Gambar 5.17 Grafik hubungan faktor utilisasi dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *bandwith CODEC* 384 Kbps 119

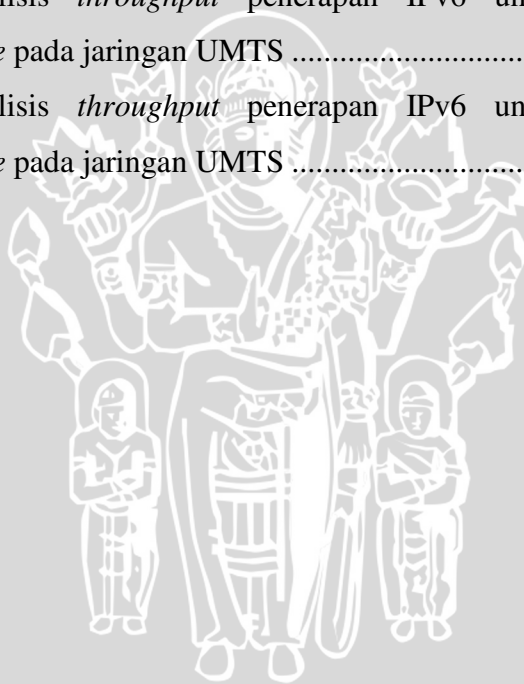
Gambar 5.18 Grafik hubungan *bandwith CODEC video* dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS 120



DAFTAR TABEL

Tabel 2.1	Audio dan video CODEC	25
Tabel 2.2	Konsumsi <i>bandwith</i> aplikasi <i>videophone</i>	27
Tabel 3.1	Macam-macam layanan UMTS dan karakteristiknya.....	34
Tabel 3.2	Tipe sel dan karakteristiknya	41
Tabel 3.3	Syarat layanan <i>Bearer</i> IMT 2000.....	41
Tabel 3.4	Parameter <i>Link Budget</i> W-CDMA.....	70
Tabel 5.1	Hasil analisis paket data aplikasi <i>videophone</i> dan <i>bandwith</i> aplikasi <i>videophone</i> berbasis IPv6 pada jaringan UMTS	77
Tabel 5.2	Hasil analisis <i>delay</i> enkapsulasi pada MT saat lintasan <i>forward</i> ...	84
Tabel 5.3	Hasil analisis <i>delay</i> enkapsulasi dan dekapsulasi pada UTRAN saat lintasan <i>forward</i>	86
Tabel 5.4	Hasil analisis <i>delay</i> enkapsulasi dan dekapsulasi pada SGSN saat lintasan <i>forward</i>	87
Tabel 5.5	Hasil analisis <i>delay</i> enkapsulasi (<i>reverse</i>) dan dekapsulasi (<i>forward</i>) pada GGSN.....	89
Tabel 5.6	Hasil analisis <i>delay</i> enkapsulasi dan dekapsulasi pada SGSN saat lintasan <i>reverse</i>	91
Tabel 5.7	Hasil analisis <i>delay</i> enkapsulasi dan dekapsulasi pada UTRAN saat lintasan <i>reverse</i>	93
Tabel 5.8	Hasil analisis <i>delay</i> dekapsulasi pada <i>Mobile Terminal (MT)</i> saat lintasan <i>reverse</i>	94
Tabel 5.9	Hasil analisis <i>delay</i> enkapsulasi dan dekapsulasi total	95
Tabel 5.10	Hasil analisis <i>delay forward</i> dan <i>reverse</i> total.....	95
Tabel 5.11	Hasil analisis <i>delay</i> transmisi total.....	100
Tabel 5.12	Hasil analisis <i>delay</i> propagasi total.....	104
Tabel 5.13	Hasil analisis <i>delay</i> antrian	106
Tabel 5.14	Hasil analisis <i>delay</i> jaringan	108
Tabel 5.15	Hasil analisis <i>delay</i> jaringan	108
Tabel 5.16	Hasil analisis <i>delay</i> jaringan	108

Tabel 5.17 Hasil analisis <i>delay end-to-end</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	109
Tabel 5.18 Hasil analisis <i>delay end-to-end</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	110
Tabel 5.19 Hasil analisis <i>delay end-to-end</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	110
Tabel 5.20 Hasil analisis probabilitas <i>packet loss</i> untuk aplikasi <i>videophone</i> .	112
Tabel 5.21 Hasil analisis probabilitas <i>packet loss total end-to-end</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	115
Tabel 5.22 Hasil analisis <i>throughput</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	118
Tabel 5.23 Hasil analisis <i>throughput</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	118
Tabel 5.24 Hasil analisis <i>throughput</i> penerapan IPv6 untuk aplikasi <i>videophone</i> pada jaringan UMTS	118



DAFTAR SINGKATAN

2G	: <i>Second Generation</i>
3G	: <i>Third Generation</i>
3GPP	: <i>3rd Generation Partnership Project</i>
AAL5	: <i>ATM Adaptation Layer 5</i>
ACF	: <i>Admission Confirm</i>
AH	: <i>Authentication Header</i>
ARQ	: <i>Admission Request</i>
ATM	: <i>Asynchronous Transfer Mode</i>
BER	: <i>Bit Error Rate</i>
BS	: <i>Base Station</i>
BSC	: <i>Base Station Controller</i>
BTS	: <i>Base Station Transceiver</i>
CDMA	: <i>Code Division Multiple Access</i>
CIF	: <i>Common Intermediate Format</i>
CN	: <i>Core Network</i>
CODEC	: <i>COder/DECoder</i>
CPCS	: <i>Common Part Convergence Sublayer</i>
CRC	: <i>Cyclic Redundancy Check</i>
CS	: <i>Circuit Switch</i>
CSRC	: <i>Contributing Source</i>
CT	: <i>CPCS Trailer</i>
EMS	: <i>Element Management System</i>
ETSI	: <i>European Telecommunication Standart Institute</i>
FDD	: <i>Frekuensi Division Duplex</i>
FIFO	: <i>First In First Out</i>
GGSN	: <i>Gateway GPRS Support Node</i>
GK	: <i>Gatekeeper</i>
G-MSC	: <i>Gateway Mobile Service Switching Centre</i>
GPRS	: <i>General Packet Radio Service</i>
GSM	: <i>Global System for Mobile Communication</i>
GTP	: <i>GPRS Tunnelling Protocol</i>
HLR	: <i>Home Location Register</i>
ICMP	: <i>Internet Control Messaging Protocol</i>
IETF	: <i>Internet Engineering Task Force</i>
IMT-2000	: <i>International Mobile Telecommunication 2000</i>
IP	: <i>Internet Protocol</i>
IPng	: <i>Next Generation Internet Protocol</i>
IPSec	: <i>Internet Protocol Security</i>
IPSec ESP	: <i>IPSec Encapsulating Security Payload</i>
IPv6	: <i>Internet Protocol version 6</i>
IPv4	: <i>Internet Protocol version 4</i>
ISDN	: <i>Integrated Service Digital Network</i>
ITU	: <i>International Telecommunication Union</i>
ITU-T	: <i>ITU Telecommunication Standardization Sector</i>
LAC	: <i>Link Access Layer</i>

MAC	: <i>Medium Access Layer</i>
MC	: <i>Multipoint Controller</i>
MCU	: <i>Multipoint Control Unit</i>
ME	: <i>Mobile Equipment</i>
MGCP	: <i>Media Gateway Control Protocol</i>
MP	: <i>Multipoint Processor</i>
MS	: <i>Mobile Station</i>
MSC	: <i>Mobile Service Switching Centre</i>
MSS	: <i>Mobile Satellite System</i>
MT	: <i>Mobile Terminal</i>
MTU	: <i>Maximum Transmission Unit</i>
NTSC	: <i>National Television Systems Committee</i>
PAL	: <i>Phase Alternating Line</i>
PC	: <i>Personal Computer</i>
PDCP	: <i>Paket Data Convergence Protocol</i>
PDN	: <i>Paket Data Network</i>
PDU	: <i>Protocol Data Unit</i>
PID	: <i>Packet Identifier</i>
PLMN	: <i>Public Line Mobile Network</i>
PS	: <i>Packet Switch</i>
QCIF	: <i>Quarter Common Intermediate Format</i>
QoS	: <i>Quality of Service</i>
RAN	: <i>Radio Access Network</i>
RGB	: <i>Red, Green, Blue</i>
RFC	: <i>Request for Comments</i>
RLC	: <i>Radio Link Control</i>
RNC	: <i>Radio Network Controller</i>
RTP	: <i>Realtime Transport Protocol</i>
SCN	: <i>Switched Circuit Network</i>
SGSN	: <i>Serving GPRS Support Node</i>
SIP	: <i>Session Initiation Protocol</i>
SSRC	: <i>Synchronization Source</i>
STM	: <i>Synchronous Transfer Module</i>
TCP	: <i>Transmission Control Protocol</i>
TDD	: <i>Time Division Duplex</i>
TDMA	: <i>Time Division Multiple Access</i>
TE	: <i>Terminal Equipment</i>
UDP	: <i>User Datagram Protocol</i>
UE	: <i>User Equipment</i>
UMTS	: <i>Universal Mobile Telecommunication System</i>
URAN	: <i>UMTS Radio Access Network</i>
USIM	: <i>UMTS Subscriber Identity Module</i>
UTRAN	: <i>UMTS Terrestrial Radio Access Network</i>
VCR	: <i>Video Cassette Recorder</i>
VLR	: <i>Visitor Location Register</i>
VoIP	: <i>Voice over Internet Protocol</i>
WARC	: <i>World Administrative Radio Conference</i>
W-CDMA	: <i>Wideband Code Division Multiple Access</i>

DAFTAR SIMBOL

$\rho_{Network}$	=	probabilitas <i>packet loss</i> pada jaringan
$\rho_{IP/UDP/RTP}$	=	probabilitas <i>packet loss</i> UDP/RTP/IP
ρ_{tot}	=	probabilitas <i>packet loss</i> total/ <i>end-to-end</i>
$\rho_{UDP/RTP/IP}$	=	probabilitas <i>packet loss</i> aplikasi <i>videophone</i>
$\rho_{network}$	=	probabilitas <i>packet loss</i> pada jaringan UMTS
ρ_{AI}	=	probabilitas <i>packet loss</i> pada <i>air interface</i> (WCDMA <i>Air interface</i>)
ρ_{UTRAN}	=	probabilitas <i>packet loss</i> pada UTRAN (Node B, RNC)
$\rho_{core\ network}$	=	probabilitas <i>packet loss</i> pada <i>core network</i> (SGSN, GGSN)
ρ_b	=	BER
ρ	=	faktor utilisasi
α	=	konstanta
λ	=	<i>throughput</i> (paket / detik)
μ	=	kecepatan pelayanan pada server (paket/detik)
μ_{gk}	=	rata-rata pelayanan di gatekeeper (paket/detik)
μ_{GGSN}	=	kecepatan pelayanan pada GGSN (paket / detik)
λ_{gk}	=	rata-rata kedatangan paket <i>gatekeeper</i> (paket/detik)
η	=	faktor pembebanan sistem
B_a	=	<i>bandwidth</i> aktual <i>audio</i> (bps)
B_{CODECV}	=	<i>bandwidth CODEC video</i> (bps)
B_{CODECA}	=	<i>bandwidth CODEC audio</i> (bps)
B_v	=	<i>bandwith</i> aktual video (bps)
B_{VP}	=	<i>bandwidth</i> aktual aplikasi <i>videophone</i>
BR	=	<i>data rate</i> (bps)
C	=	kecepatan standar <i>gatekeeper</i> (bps)
CRC	=	<i>Cyclic Redundancy Check</i> (byte)
c	=	cepat rambat gelombang (m/detik)
C_f	=	faktor aktivitas kanal suara/data
C_{GGSN}	=	kapasitas link pada GGSN (bps)

C_{trans}	=	kapasitas saluran transmisi (bps)
d	=	jarak (m)
E_b/N_o	=	signal to noise ratio (dB)
$EIRP_{MS}$	=	EIRP mobile station (dBm)
Frame rate	=	waktu pembentukan tiap frame (detik)
G_{MS}	=	gain antenna mobile station (dBi)
G_{BS}	=	gain antenna base station (dBi)
$H_{UDP/RTP/IP}$	=	header UDP/RTP/IP (byte)
Header IPv6	=	panjang header IPv6 (byte)
Header ICMP	=	panjang header ICMP (byte)
Header PDCP	=	header PDCP (byte)
Header RLC	=	header RLC (byte)
Header MAC	=	header MAC (byte)
Header GTP	=	header GTP (byte)
Header UDP	=	header UDP (byte)
Header IP	=	header IPv6 (byte)
Header PAD	=	header PAD (byte)
Header CT	=	header CT (byte)
H_{frame}	=	panjang header frame PDU UMTS (byte)
h_1	=	jumlah hop pada jaringan UTRAN (Node B, RNC)
h_2	=	jumlah hop pada jaringan core network (SGSN, GGSN)
I_m	=	interference margin (dB)
l	=	panjang paket (byte)
l'	=	panjang header IPv6 (byte)
L_{MS}	=	rugi kabel, konektor, dan kombiner (dB)
L_P	=	rugi-rugi propagasi (dB)
L_f	=	loss feeder (dB)
m	=	jumlah paket data pada GGSN (paket)
MTU_{PDCP}	=	Maximum Transmission Unit PDCP (byte)
MTU_{GTP}	=	Maximum Transmission Unit GTP (byte)
$MTU_{Ethernet}$	=	Maximum Transmission Unit Ethernet (byte)
$N_{frame-PDCP}$	=	jumlah frame PDCP

$N_{frame-RLC}$	=	jumlah <i>frame</i> RLC PDU
$N_{frame GTP}$	=	jumlah <i>frame</i> GTP
$N_{frame ATM}$	=	jumlah <i>frame</i> <i>payload</i> ATM
$N_{frame SGSN}$	=	jumlah <i>Frame</i> SGSN
$N_{frame GGSN}$	=	jumlah <i>Frame</i> GGSN
n	=	jumlah slot tiap MS
N_0	=	<i>thermal noise</i> (dBm/Hz)
NF_{BS}	=	<i>noise figure</i> base station (dB)
NF_{MS}	=	<i>noise figure</i> mobile station (dB)
N_t	=	jumlah kanal trafik pada sektor sel
N_{frame}	=	jumlah <i>frame</i> yang ditransmisikan dari sumber ke tujuan
P	=	ukuran paket data di <i>gatekeeper</i> (bit)
P_a	=	jumlah paket <i>audio</i> yang dihasilkan tiap detik (paket per detik)
P_{a-size}	=	besar tiap paket <i>audio</i> (bit)
PL_a	=	<i>payload</i> tiap paket <i>audio</i> (bit)
PL_v	=	<i>payload</i> tiap paket <i>video</i> (bit)
PL_{VP}	=	<i>payload</i> <i>videophone</i> (byte)
P_{size}	=	ukuran paket data (byte)
$P_{vp-size}$	=	ukuran paket aplikasi <i>videophone</i> (bit)
P_e	=	probabilitas bit <i>error</i> (BER)
P_{cu}	=	daya penerimaan <i>base station</i> tiap <i>user</i> (dBm)
pt	=	ERP kanal trafik (dBm)
P_t	=	ERP seluruh kanal trafik dari <i>base station</i> (dBm)
p_{tr}	=	daya yang diterima kanal trafik (dBm)
P_{TX-MS}	=	daya transmisi <i>mobile station</i> (dBm)
PL_{frame}	=	<i>payload</i> <i>frame</i> PDU UMTS (byte)
R	=	radius cell (m)
S	=	jarak antara MT dengan Node B/radius sel (m)
T_{trx}	=	waktu pentransmisian dari <i>media server</i> (detik)
t_{CODEC}	=	<i>delay</i> CODEC aplikasi <i>videophone</i> (detik)
t_a	=	<i>delay</i> CODEC <i>audio</i> (detik)
t_{gk}	=	<i>delay</i> <i>gatekeeper</i> (detik)

t_{net}	=	<i>delay jaringan (detik)</i>
t_{tot}	=	<i>delay end-to-end aplikasi videophone (detik)</i>
t_v	=	<i>delay CODEC video (detik)</i>
t_{proc}	=	<i>delay proses (detik)</i>
t_w	=	<i>delay antrian (detik)</i>
t_T	=	<i>delay transmisi (detik)</i>
t_p	=	<i>delay propagasi (detik)</i>
t_{E1}	=	<i>delay enkapsulasi di MT saat lintasan forward (dtk)</i>
t_{E2}	=	<i>delay enkapsulasi di UTRAN saat lintasan forward (dtk)</i>
t_{E3}	=	<i>delay enkapsulasi di SGSN saat lintasan forward (dtk)</i>
t_{E4}	=	<i>delay enkapsulasi di GGSN saat lintasan reverse (dtk)</i>
t_{E5}	=	<i>delay enkapsulasi di SGSN saat lintasan reverse (dtk)</i>
t_{E6}	=	<i>delay enkapsulasi di UTRAN saat lintasan reverse (dtk)</i>
t_{D1}	=	<i>delay dekapsulasi di UTRAN saat lintasan forward (detik)</i>
t_{D2}	=	<i>delay dekapsulasi di SGSN saat lintasan forward (detik)</i>
t_{D3}	=	<i>delay dekapsulasi di GGSN saat lintasan forward (detik)</i>
t_{D4}	=	<i>delay dekapsulasi di SGSN saat lintasan reverse (detik)</i>
t_{D5}	=	<i>delay dekapsulasi di UTRAN saat lintasan reverse (detik)</i>
t_{D6}	=	<i>delay dekapsulasi di MT saat lintasan reverse (detik)</i>
t_{T1}	=	<i>delay transmisi pada Uu interface lintasan forward (detik)</i>
t_{T2}	=	<i>delay transmisi pada Iub interface lintasan forward (detik)</i>
t_{T3}	=	<i>delay transmisi pada Iu Interface saat lintasan forward (detik)</i>
t_{T4}	=	<i>delay transmisi saat lintasan forward pada Gn interface (detik)</i>
t_{T5}	=	<i>delay transmisi pada Gn interface saat lintasan reverse (detik)</i>
t_{T6}	=	<i>delay transmisi pada Iu interface saat lintasan reverse (detik)</i>
t_{T7}	=	<i>delay transmisi pada Iub interface lintasan reverse (detik)</i>
t_{T8}	=	<i>delay transmisi pada Uu interface lintasan reverse (detik)</i>
$t_{T forward}$	=	<i>delay transmisi untuk lintasan forward (detik)</i>

$t_{T\ reverse}$	=	<i>delay</i> transmisi untuk lintasan <i>reverse</i> (detik)
$t_{T\ Total}$	=	<i>delay</i> transmisi total (detik)
t_{p1}	=	<i>delay</i> propagasi dari MT ke Node B saat lintasan <i>forward</i> (detik)
t_{p2}	=	<i>delay</i> propagasi dari Node B ke RNC saat lintasan <i>forward</i> (detik)
t_{p3}	=	<i>delay</i> propagasi dari RNC ke SGSN saat lintasan <i>forward</i> (detik)
t_{p4}	=	<i>delay</i> propagasi dari SGSN ke GGSN saat lintasan <i>forward</i> (detik)
t_{p5}	=	<i>delay</i> propagasi dari GGSN ke SGSN saat lintasan <i>reverse</i> (detik)
t_{p6}	=	<i>delay</i> propagasi dari SGSN ke RNC saat lintasan <i>reverse</i> (detik)
t_{p7}	=	<i>delay</i> propagasi dari RNC ke Node B saat lintasan <i>reverse</i> (detik)
t_{p8}	=	<i>delay</i> propagasi dari Node B ke MT saat lintasan <i>reverse</i> (detik)
$t_{P\ forward}$	=	<i>delay</i> propagasi untuk lintasan <i>forward</i> (detik)
$t_{P\ reverse}$	=	<i>delay</i> propagasi untuk lintasan <i>reverse</i> (detik)
$t_{P\ Total}$	=	<i>delay</i> propagasi total (detik)
t_q	=	waktu tunggu paket dalam saluran (detik)
t_s	=	rata-rata kecepatan pelayanan paket pada GGSN (detik)
t_w	=	<i>delay</i> antrian (detik)
t_v	=	waktu transmisi rata-rata sebuah paket yang diterima dalam keadaan benar (detik)
t_l	=	waktu yang dibutuhkan untuk mentransmisikan sebuah paket melalui jaringan UMTS (detik)
t_T	=	waktu total yang dibutuhkan untuk mentransmisikan sebuah paket (detik)
V	=	kecepatan propagasi informasi pada media/saluran transmisi (meter/detik)

V_{MT}	=	kecepatan transmisi data di MT (bps)
V_{UTRAN}	=	kecepatan transmisi data di UTRAN (bps)
V_{SGSN}	=	kecepatan transmisi data di SGSN (bps)
V_{GGSN}	=	kecepatan transmisi data di GGSN (bps)
V_{MT}	=	kecepatan <i>data rate</i> (bps)
$V_{ethernet}$	=	kecepatan transmisi pada <i>Gn interface</i> (bps)
$V_{Iu\ interface}$	=	kecepatan transmisi pada <i>Iu interface</i> (bps)
$V_{Iub\ interface}$	=	kecepatan transmisi pada <i>Iub Interface</i> (bps)
W_{AAL5}	=	besar paket AAL5 (byte)
$W_{total\ frame\ ATM}$	=	jumlah <i>total frame</i> ATM
$W_{datagram}$	=	panjang <i>datagram</i> IP (byte)
W_{segmen}	=	panjang segmen TCP/UDP (byte)
W_{data}	=	jumlah <i>application data</i> /data sebelum terenkapsulasi (byte)
W_{PDCP}	=	panjang data PDCP (byte)
$W_{frame\ PDCP}$	=	panjang <i>frame</i> PDCP (byte)
$W_{frame\ PDCP\ total}$	=	panjang <i>total frame</i> pada PDCP
$W_{frame\ RLC}$	=	panjang <i>frame</i> RLC PDU (byte)
$W_{total\ frame\ RLC}$	=	jumlah <i>total frame</i> RLC PDU (byte)
$W_{frame\ MAC}$	=	panjang <i>frame</i> MAC PDU (byte)
$W_{total\ frame\ MAC}$	=	jumlah <i>total frame</i> MAC PDU (byte)
$W_{frame\ GTP}$	=	panjang <i>frame</i> GTP (byte)
W_{SGSN}	=	panjang paket SGSN (byte)
$W_{frame\ SGSN}$	=	panjang <i>frame</i> SGSN
$W_{total\ frame\ SGSN}$	=	jumlah <i>total frame</i> SGSN
$W_{frame\ GGSN}$	=	panjang <i>frame</i> GGSN
$W_{total\ frame\ GGSN}$	=	jumlah <i>total frame</i> GGSN

BAB I

PENDAHULUAN

1.1 Latar Belakang

Meningkatnya kebutuhan masyarakat akan sarana telekomunikasi yang *real time*, praktis, serta mampu mengatasi masalah ruang yang cakupannya sangat luas bahkan sulit terjangkau, merupakan faktor utama dalam kemajuan teknologi telekomunikasi saat ini. Atas dasar inilah yang mendorong pesatnya perkembangan dalam sistem komunikasi bergerak hingga saat ini. *Universal Mobile Telecommunication System (UMTS)* merupakan generasi ketiga dari teknologi *wireless* yang menjadi standar sistem komunikasi bergerak dan diorganisasikan pada tingkat global yang distandarkan oleh ETSI (*European Telecommunication Standart Institute*) sebagai jawaban atas semakin tingginya kapasitas dan kualitas layanan yang dibutuhkan.

IPv6 (*Internet Protocol version 6*) yang disebut juga sebagai *Next Generation Internet Protocol (IPng)* direkomendasikan oleh Direktur Area IPng dari IETF (*Internet Engineering Task Force*) dalam RFC 1752. IPv6 merupakan teknik pengalamatan yang baru yang memberikan jangkauan alamat yang lebih luas dibandingkan dengan IPv4. Sehingga diharapkan dengan adanya IPv6 mampu mengatasi keterbatasan alamat, *routing* serta *auto* konfigurasi jaringan pada IPv4.

Videophone merupakan salah satu teknologi multimedia yang dapat menyalurkan informasi suara dan gambar dengan menggunakan jaringan *Internet Protokol (IP)* sebagai media transmisinya. Aplikasi *videophone* merupakan salah satu aplikasi yang bisa diterapkan pada sistem jaringan UMTS. Teknologi *videophone* pada jaringan UMTS mempunyai beberapa model pemanggilan yang dapat dilakukan yaitu antar *Mobile Station (MS)*, antar MS dengan pesawat *videophone*, antar MS dengan *Personal Computer (PC)* maupun dengan PC.

Penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS diharapkan nantinya dapat menjanjikan pemanfaatan sumber daya jaringan yang lebih efisien dengan kualitas layanan tertentu. Oleh karena itu sangatlah perlu dilakukan analisis/perhitungan untuk melihat bagaimana performansi jaringan

melalui pendekatan *delay end-to-end*, *packet loss*, dan *throughput* sebagai bahan pertimbangan dan masukan untuk perkembangan layanan multimedia di masa yang akan datang.

1.2 Rumusan Masalah

Dengan memperhatikan latar belakang dari skripsi ini, maka perumusan masalah dititik beratkan pada :

1. Bagaimana konsep dasar IPv6 ?
2. Bagaimana konsep dasar teknologi aplikasi *videophone* ?
3. Bagaimana konfigurasi jaringan paket data UMTS ?
4. Bagaimana konfigurasi penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS ?
5. Bagaimana analisis perhitungan performansi penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS yang meliputi *delay end-to-end*, *packet loss*, dan *throughput* ?
6. Bagaimana pengaruh konsumsi *bandwith CODEC* aplikasi *videophone* terhadap performansi jaringan UMTS yang berbasiskan IPv6 ditinjau dari *delay end-to-end*, *packet loss*, dan *throughput* ?

1.3 Ruang Lingkup

Pada penulisan skripsi ini dibatasi pada masalah :

1. Model hubungan yang digunakan dalam analisis adalah antara *Mobile Terminal* pengirim dengan *Mobile Terminal* penerima.
2. Pembahasan tentang UMTS berdasarkan pada standar 3GPP (*3rd Generation Partnership Project*).
3. Metode *multiple acces* yang digunakan ialah FDD W-CDMA.
4. Aplikasi *videophone* menggunakan protokol H.323 dengan jenis *CODEC* H.263 untuk *video* dan G.723.1 untuk *voice* dengan *frame rate* 30 ms.
5. *Internet Protocol* yang dipergunakan adalah IPv6.
6. Tidak membahas tentang pengalamatan IPv6.
7. Tidak membahas perangkat keras dan lunak sistem.
8. Tidak membahas mengenai teknik modulasi, teknik pensinyalan, *handoff*, dan teknik multipleksing.

1.4 Tujuan

Penulisan skripsi ini bertujuan untuk menganalisis penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.

1.5 Sistematika Penulisan

Pada penulisan skripsi ini terdiri dari 6 bab yang secara garis besar dapat dijabarkan sebagai berikut :

BAB I Pendahuluan, memuat tentang latar belakang, rumusan masalah, ruang lingkup, tujuan dan sistematika penulisan.

BAB II Internet Protokol Versi 6 dan *Videophone*, merupakan studi pustaka tentang IPv6 dan *Videophone*, konsep dasar tentang IPv6 dan *Videophone*.

BAB III Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS, merupakan studi pustaka tentang UMTS yang difokuskan pada konsep dasar, penjelasan tentang definisi, konfigurasi jaringan dan konsep pelayanan pada UMTS secara umum, serta persamaan untuk menghitung parameter performansi penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.

BAB IV Metodologi Penelitian, merupakan penjelasan tentang metode yang digunakan dalam menyusun skripsi ini.

BAB V Analisis Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS, membahas tentang analisis penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.

BAB VI Penutup, memuat kesimpulan dari semua hal yang telah dibahas dan diuraikan sebelumnya, serta saran.





BAB II

INTERNET PROTOKOL VERSI 6 (IPv6) DAN VIDEOPHONE

2.1 INTERNET PROTOKOL VERSI 6 (IPv6)

2.1.1 Umum

IP versi 6 (IPv6) adalah protokol internet versi baru yang didesain sebagai pengganti dari internet protokol versi 4 (IPv4). Perubahan dari IPv4 ke IPv6 pada dasarnya terjadi karena beberapa hal yang dikelompokkan dalam kategori berikut:

➤ Kapasitas Perluasan Alamat

IPv6 meningkatkan ukuran dan jumlah alamat yang mampu didukung oleh IPv4 dari 32 bit menjadi 128 bit. Peningkatan kapasitas alamat ini digunakan untuk mendukung peningkatan hirarki atau kelompok pengalamatan, peningkatan jumlah atau kapasitas alamat yang dapat dialokasikan dan diberikan pada *node*, dan mempermudah konfigurasi alamat pada *node* sehingga dapat dilakukan secara otomatis. Peningkatan skalabilitas juga dilakukan pada *routing multicast* dengan meningkatkan cakupan dan jumlah pada alamat *multicast*. IPv6 ini selain meningkatkan jumlah kapasitas alamat yang dapat dialokasikan pada *node*, juga mengenalkan jenis atau tipe alamat baru, yaitu alamat *anycast*. Tipe alamat *anycast* ini didefinisikan dan digunakan untuk mengirimkan paket ke salah satu dari kumpulan *node*.

➤ Penyederhanaan Format *Header*

Beberapa kolom pada *header* IPv4 telah dihilangkan atau dapat dibuat sebagai *header* pilihan. Hal ini digunakan untuk mengurangi biaya pemrosesan hal-hal yang umum pada penanganan paket IPv6 dan membatasi biaya *bandwidth* pada *header* IPv6. Dengan demikian pemrosesan *header* pada paket IPv6 dapat dilakukan secara efisien.

➤ Peningkatan Dukungan untuk *Header* Pilihan dan *Header* Tambahan (*option and extension header*)

Perubahan yang terjadi pada *header-header* IP yaitu dengan adanya pengkodean *header options* (pilihan) pada IP dimaksudkan agar lebih efisien dalam penerusan paket (*packet forwarding*), agar tidak terlalu ketat dalam pembatasan panjang *header* pilihan yang terdapat dalam paket IPv6 dan sangat fleksibel dimungkinkan untuk mengenalkan *header* pilihan baru pada masa akan datang.

➤ Kemampuan pelabelan aliran paket

Kemampuan atau fitur baru yang ditambahkan pada IPv6 ini adalah memungkinkan pelabelan paket atau pengklasifikasian paket yang merupakan milik dari aliran trafik tertentu dan aliran trafik ini meminta penanganan spesial, seperti kualitas mutu layanan tertentu (QoS) atau layanan *real-time*.

➤ Autentifikasi dan Kemampuan Privasi

Kemampuan tambahan untuk mendukung autentifikasi, integritas data dan data penting juga dispesifikasikan dengan alamat IPv6.

2.1.2 Konfigurasi Protokol TCP/IP

Model protokol TCP/IP yang digunakan pada penulisan skripsi ini adalah model internet empat lapis yang terdiri dari sekumpulan protokol yang masing-masing bertanggungjawab atas bagian-bagian tertentu dari komunikasi data.

Keempat lapis / *layer* tersebut adalah :

- *Network Access Layer*
- *Internet Layer*
- *Host to host Transport Layer*
- *Application Layer*

2.1.2.1 *Network Access Layer*

Lapisan Akses *Network* bertanggung jawab untuk pertukaran data antara suatu *host* dan jaringan, serta untuk pengiriman data antara dua peralatan pada jaringan yang sama. Beberapa contohnya adalah Ethernet, ATM, ISDN, dll.



Proses / Aplikasi	<table border="1"> <tr> <td>FTF</td> <td>TELNET</td> <td>SMTP</td> <td>Other</td> <td>TFTP</td> <td>DNS</td> <td>SNMP</td> <td>NFS</td> </tr> </table>								FTF	TELNET	SMTP	Other	TFTP	DNS	SNMP	NFS
FTF	TELNET	SMTP	Other	TFTP	DNS	SNMP	NFS									
Host ke Host	TCP				UDP											
Internet	Protokol Routing	ARP RARP	IP				ICMP									
Akses Jaringan	Ethernet II, IEEE 802.x, FDDI, ATM, ISDN, X.25, dsb															

Gambar 2.1 — Protokol pada tumpukan protokol TCP/IP
 Sumber : D. Heywood, 1997 : 144

2.1.2.2 Internet Layer

Lapisan internet bertanggung jawab untuk pengiriman data melalui antar jaringan. Protokol lapisan internet yang utama adalah *Internet Protocol (IP)* yang bertanggung jawab untuk lapisan tersebut.

IP menggunakan protokol-protokol lain untuk tugas-tugas khusus. *Internet Control Messaging Protocol (ICMP)* digunakan untuk mengirimkan pesan-pesan ke lapisan *host to host*. IP merupakan protokol *internet* yang diperlukan yang fungsinya adalah sebagai berikut:

- Pengalamatan
- Fragmentasi datagram pada antar jaringan
- Pengiriman datagram pada antar jaringan

2.1.2.3 Host to host Transport Layer

Lapisan *host to host* memiliki dua macam fungsi utama sebagai berikut :

- Menjadi *interface* yang baik antara proses-proses dan aplikasi lapisan atas ke jaringan.
- Mengirimkan pesan lapisan atas antara *host*.

Karena proses lapisan atas memiliki kebutuhan yang bervariasi, terdapat dua protokol *host to host* yaitu *Transmission Control Protocol (TCP)* dan *User Datagram Protocol (UDP)*. Pada bagian ini yang dibahas adalah TCP sedangkan UDP akan di bahas pada bagian *videophone*.

TCP (*Transmission Control Protocol*)

TCP merupakan protokol yang menyediakan layanan bervariasi yaitu : *connection oriented, reliable, byte stream service*. *Connection oriented* berarti sebelum melakukan pertukaran data, dua aplikasi pengguna TCP harus melakukan pembentukan hubungan (*handshake*) terlebih dahulu. *Reliable* berarti TCP menerapkan proses deteksi kesalahan paket dan retransmisi. *Byte stream service* berarti paket dikirimkan ke tujuan secara berurutan.

Agar reliabilitas pengiriman data terjamin, TCP melakukan hal-hal berikut :

- Data yang diterima oleh aplikasi dipecah menjadi segmen-segmen yang besarnya menurut TCP paling sesuai untuk mengirimkan data.
- Ketika TCP menerima data dari mitranya, TCP mengirimkan *acknowledgment* (pemberitahuan bahwa ia telah menerima data).
- Ketika TCP mengirimkan sebuah data, TCP mengaktifkan pewaktu (*software timer*) yang akan menunggu *acknowledgment* dari penerima segmen data tersebut. Jika sampai waktu yang ditentukan tidak diterima *acknowledgment*, data tersebut dikirimkan kembali oleh TCP.
- Sebelum segmen data dikirim, TCP melakukan perhitungan *checksum* pada *header* dan datanya. Hal ini berbeda dengan protokol IP yang hanya melakukan perhitungan *checksum* pada *headernya* saja. Jika segmen yang diterima memiliki *checksum* yang tidak valid, TCP akan membuang segmen ini dan berharap sisi pengirim akan melakukan retransmisi.
- Karena segmen TCP dikirim menggunakan IP, dan datagram IP dapat sampai ke tujuan dalam keadaan tidak berurutan, segmen TCP yang dikirimkan pun dapat mengalami hal yang sama. Karenanya sisi penerima paket TCP harus mampu melakukan pengurutan kembali segmen TCP yang ia terima (*resequencing*), dan memberikan data dengan urutan yang benar ke aplikasi penggunaannya.
- Karena paket IP dapat terduplikasi di perjalanan, penerima TCP harus membuang data tersebut.
- Untuk mencegah agar *server* yang cepat tidak membanjiri *server* yang lambat, TCP melakukan proses *flow control*. Setiap koneksi TCP memiliki *buffer* dengan ukuran yang terbatas. Sisi penerima TCP hanya

memperbolehkan sisi pengirim mengirimkan data sebesar *buffer* yang ia miliki.

Source Port		Destination Port	
Sequence Number			
Acknowledgment Number			
hdr	Resv	Control	Window
Checksum		Urgent Pointer	
TCP Options			
Application Data			

Gambar 2.2 Format segmen TCP
 Sumber : Purbo, 2001 : .54

Seperti yang ditunjukkan oleh Gambar 2.2, segmen TCP terdiri atas beberapa *field*. *Source* dan *destination port* adalah *field* berisi angka yang mengidentifikasi aplikasi pengirim dan penerima segmen TCP ini. *Sequence number* berisi nomor urut *byte stream* dalam data aplikasi yang dikirim. Setiap kali data ini sukses dikirim, pihak penerima data mengisi *field acknowledgment number* dengan *sequence number* berikutnya yang diharapkan penerima.

Header length berisi panjang *header* TCP. Dengan lebar 4 bit, *field* ini harus merepresentasikan panjang *header* TCP dalam satuan 4 byte.

Field window diisi dengan panjang *window* (semacam *buffer*) penerimaan segmen TCP, merupakan banyak byte maksimal yang bisa diterima setiap saat.

2.1.2.4 Application Layer

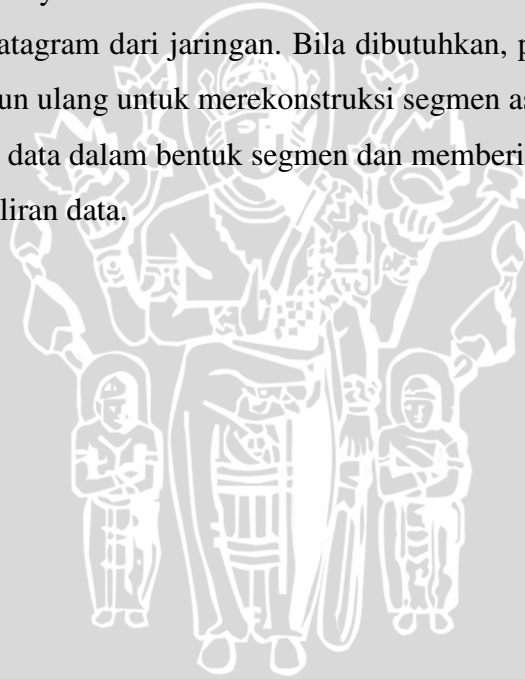
Lapisan proses/aplikasi adalah tempat dilakukannya proses kerja yang sesungguhnya. Disini dapat ditemukan program yang menyediakan pelayanan jaringan, seperti *mail server*, *file transfer server*, *remote terminal* dan *system management server*, termasuk juga program yang berinteraksi langsung dengan user seperti *ftp* dan *telnet*

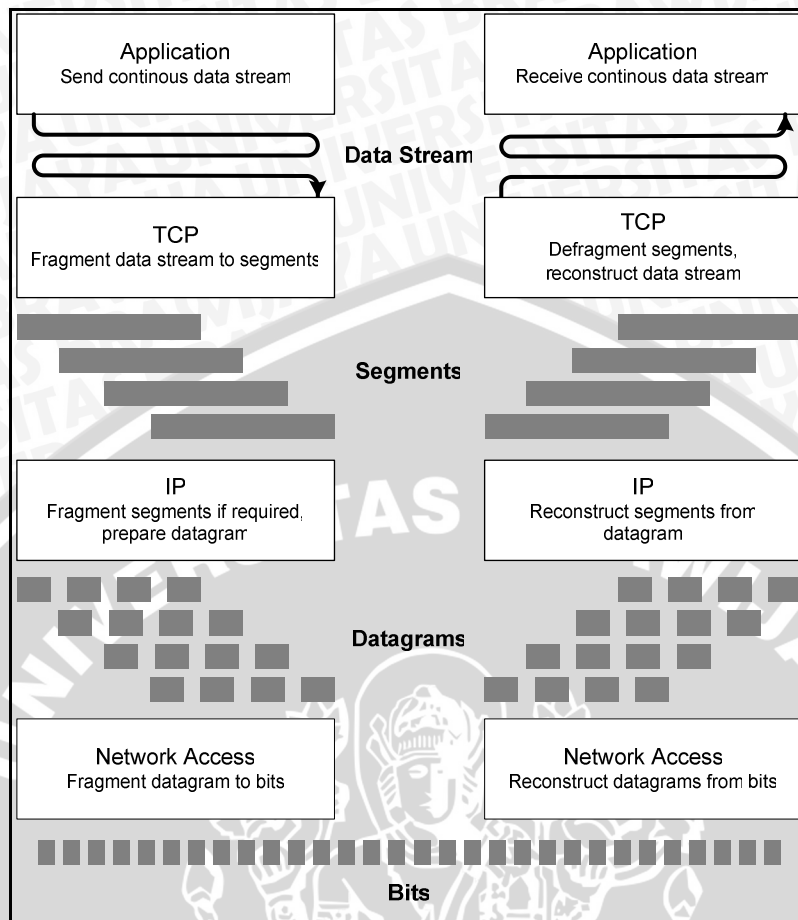
2.1.3 Pemrosesan Aliran Data

Dari sudut pandang proses, komunikasi dengan jaringan melibatkan pengiriman dan penerimaan aliran data secara berkesinambungan. Proses tidak

bertanggung jawab untuk melakukan fragmentasi data agar sesuai dengan protokol lapisan bawah. Gambar 2.3 menggambarkan bagaimana data diproses sambil berjalan menuruni tumpukan protokol, melalui jaringan, dan naik melewati tumpukan protokol pada penerima yang dapat dijelaskan sebagai berikut :

1. TCP menerima aliran data dari proses lapisan atas.
2. TCP dapat melakukan fragmentasi aliran data menjadi segmen yang memenuhi aturan ukuran maksimum datagram pada IP.
3. IP dapat melakukan fragmentasi segmen sambil mempersiapkan datagram yang ukurannya disesuaikan untuk memenuhi batasan dari jaringan.
4. Protokol jaringan mengirimkan datagram dalam bentuk bit.
5. Protokol jaringan pada *host* penerima menyusun ulang datagram dari bit-bit yang diterimanya.
6. IP menerima datagram dari jaringan. Bila dibutuhkan, potongan-potongan datagram disusun ulang untuk merekonstruksi segmen asalnya.
7. TCP menerima data dalam bentuk segmen dan memberikan ke lapisan atas dalam bentuk aliran data.



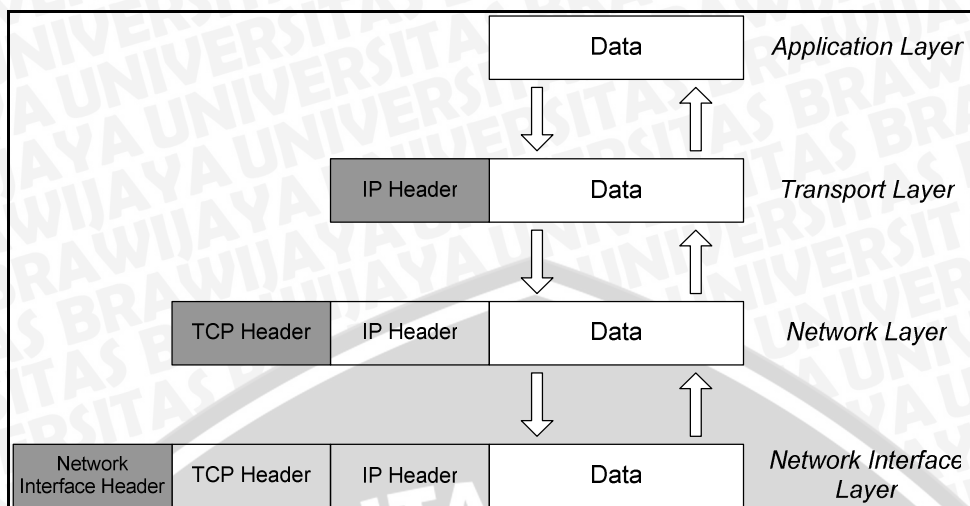


Gambar 2.3 Pemrosesan data selama pengiriman dan penerimaan
 Sumber : D. Heywood, 1997 : 148

Dalam TCP/IP, terjadi penyampaian data dari protokol yang berada di satu *layer* ke protokol yang berada di *layer* yang lain. Setiap protokol memperlakukan semua informasi yang diterimanya dari protokol lain sebagai data seperti yang di gambarkan pada gambar 2.4.

Jika suatu protokol menerima data dari protokol lain di *layer* atasnya, ia akan menambahkan informasi tambahan miliknya yang disebut *header* ke data tersebut. Informasi ini memiliki fungsi yang sesuai dengan fungsi protokol tersebut. Setelah itu, data ini diteruskan lagi ke protokol pada *layer* di bawahnya.

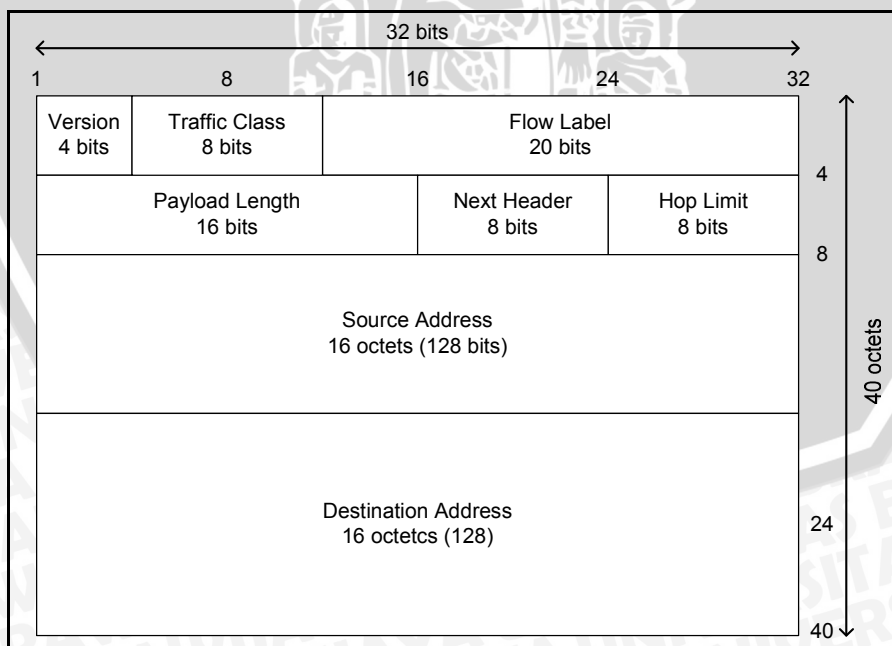
Hal yang sebaliknya terjadi jika suatu protokol menerima data dari protokol lain yang berada pada *layer* dibawahnya. Jika data ini di anggap valid, protokol akan melepas informasi tambahan atau *header* tersebut, untuk kemudian meneruskan data itu ke protokol lain yang berada pada *layer* diatasnya.



Gambar 2.4 Pergerakan data dalam *layer* TCP/IP
 Sumber : Purbo, 2001 : 2

2.1.4 Format *Header* IPv6

Format *header* alamat IPv6 menyederhanakan format *header* pada alamat IPv4. Format *header* IPv6 diperlihatkan oleh gambar 2.5. *Header* IPv6 didesain mempunyai panjang *header* 40 *byte* dan memiliki *field* lebih sedikit dibandingkan dengan IPv4, sehingga *router* melakukan penanganan tiap *field* lebih sedikit menyebabkan proses penjaluran atau aliran data menjadi lebih ringkas.



Gambar 2.5 Format *header* IPv6
 Sumber :Syarvianti, 2005 : 19

Setiap paket IPv6 membawa data yang terdiri dari :

- *Version* (4 bit), berisi nomor versi protokol internet, yaitu 6.
- *Traffic Class* (8 bit), berisi kolom kelas atau klasifikasi trafik. Tersedia untuk digunakan oleh permulaan *node* atau *router forwarding* untuk mengidentifikasi dan membedakan antara kelas yang berbeda atau prioritas paket IPV6.
- *Flow Label* (20 bit), berisi label aliran dari trafik. Digunakan oleh *host* untuk melabel paket-paket yang mana meminta penanganan khusus oleh *router* dalam suatu jaringan.
- *Payload Length* (16 bit), merupakan panjang *payload*/muatan IPv6. *Payload* ini adalah sisa paket setelah *header* IPv6 dalam bentuk oktet (di luar *header* IPv6). Catatan bahwa *header* tambahan/*extention header* yang ada dihitung sebagai bagian dari *payload* termasuk panjang *header* tambahan tersebut.
- *Next Header* (8 bit), merupakan identifikasi tipe *header* yang akan ada setelah *header* IPv6. Nilai *header* ini menggunakan tipe *header* yang sama dengan yang ada pada protokol IPv4.
- *Hop Limit* (8 bit), nilai pada kolom ini akan dikurangi satu jika paket ini melewati *node* yang berfungsi melewati/memforward paket (melewati *router*). Paket ini akan dibuang jika batas *hop* ini berubah menjadi nol.
- *Source Address* (128 bit), merupakan alamat IPv6 asal dari paket.
- *Destination Address* (128 bit), merupakan alamat IPv6 tujuan paket.

2.1.5 *Extension Header IPv6*

Extension header merupakan *header* pilihan pada IPv6. Masing-masing *extension header* panjangnya 8 *byte* (64 bit). Seluruh *extension header* secara bersama-sama membentuk daftar rangkaian *header*. *Extension header* tersedia untuk sejumlah fungsi yang berbeda, tetapi tidak dianggap sebagai bagian *header* IPv6 itu sendiri.

Extension Header meliputi:

➤ ***Hop By Hop Header (Lompatan demi Lompatan)***

Header ini membawa informasi tambahan (bila ada) serta harus diproses oleh tiap *node* dan *router* sepanjang jalur pengiriman. *Header* ini terletak tepat setelah IPv6 *base header* (*header* yang selalu diberikan pada IPv6).

➤ ***Destination Option Header***

Membawa informasi pilihan yang secara khusus ditujukan untuk alamat tujuan paket.

➤ ***Routing Header***

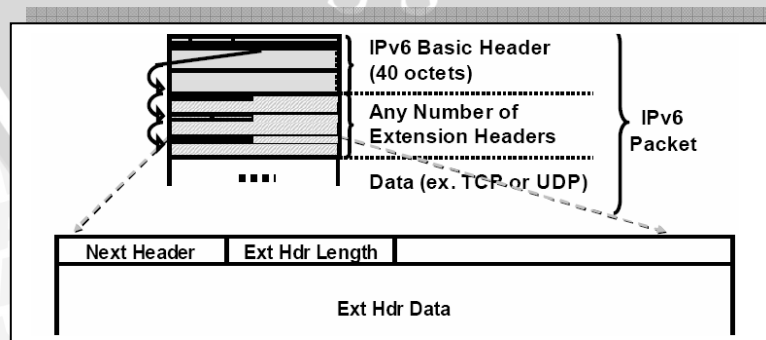
Routing header mendorong proses *routing* melalui sejumlah *intermediate router*. *Header* ini berisi daftar *node* yang harus dilewati dalam perjalanan menuju tujuan. Masing-masing *node* dalam daftar memproses *header* dan mengubah alamat tujuan ke *router* selanjutnya dalam daftar. *Header* ini serupa dengan *option* “*Loose Source Route*” dalam IPv4.

➤ ***Fragment Header***

Digunakan oleh sumber ketika paket difragmentasi, di mana paket dengan ukuran lebih besar akan dipecah sesuai dengan *path MTU (Maximum Transmission Unit)*. Fragmentasi hanya dilakukan oleh sumber paket, bukan *router*. posisi fragmen tertentu dalam paket asli dan digunakan oleh *node* tujuan untuk menyusun kembali paket.

➤ ***IPSec Authentication Header***

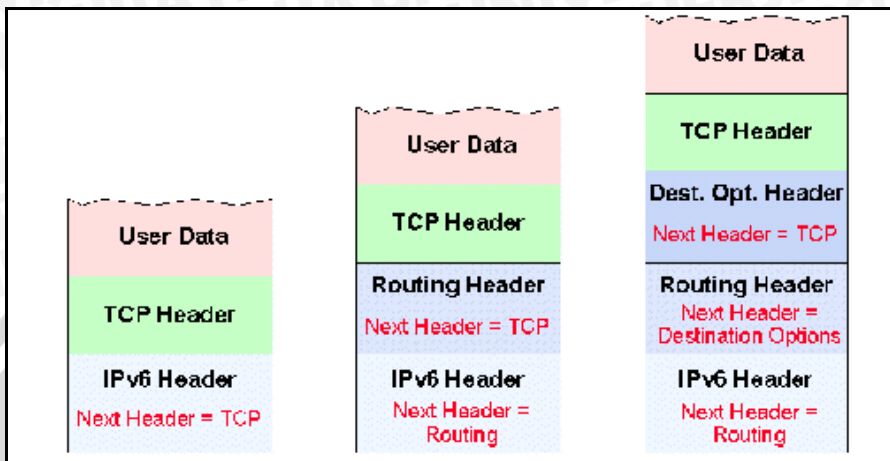
IPSec Authentication Header (AH) menyediakan: kerahasiaan, integritas, dan autentikasi sumber.



Gambar 2.6 Format extension header pada paket IPv6
Sumber: www.IPv6 Indonesia.com, 2002

➤ **IPSec ESP Header**

Header ini menyediakan layanan yang sama seperti *Authentication Header*, tetapi menambahkan fungsi enkripsi. (proses perubahan data dalam bentuk sandi untuk keamanan informasi).



Gambar 2.7 Contoh pemakaian *extension header* pada IPv6

2.1.6 Maximum Transmission Unit (MTU) IPv6 /ukuran paket IPv6

Maximum Transmission Unit IPv6 adalah ukuran maksimum paket IPv6 sebelum menuju lapisan *interface*. Besarnya MTU IPv6 tidak kurang dari 1280 byte dan dapat lebih besar tergantung jenis lapisan *interface* yang digunakan. Misalnya jika lapisan *interface* menggunakan *Ethernet* maka besarnya MTU adalah 1500 byte.

2.1.7 Proses Enkapsulasi dan Dekapsulasi

Dalam pengiriman data dari satu *node* ke *node* lain yang melewati peralatan transmisi data, pasti melalui proses enkapsulasi dan dekapsulasi paket. Proses enkapsulasi adalah proses yang terjadi pada pengirim ketika suatu data mendapatkan *header-header* tambahan sebagai informasi kontrol. Proses dekapsulasi adalah proses yang terjadi pada sisi penerima, yaitu pada saat data bergerak ke layer-layer selanjutnya melalui susunan protokol dan setiap layer membuka dan mengambil headernya dari unit data (data yang terdiri dari informasi dan *header*).

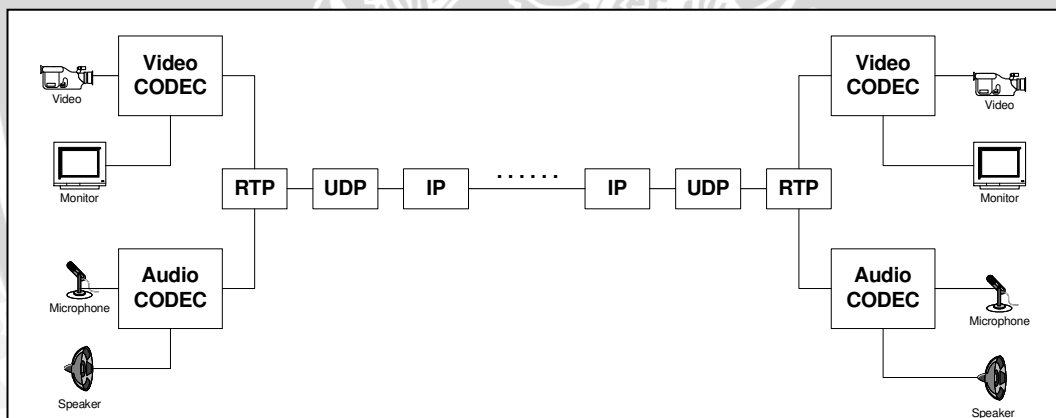


2.2 VIDEOPHONE

2.2.1 Umum

Videophone adalah salah satu bentuk *tele video conference* yang menggabungkan beberapa komunikasi yaitu suara, gambar dan data. Pada sistem konferensi *video* ini menjanjikan komunikasi yang lebih cepat dalam penyampaian informasi, lebih efisien dan lebih baik. Informasi bergambar ini sangat efektif dalam komunikasi seperti untuk keperluan bisnis, kedokteran, penelitian, maupun untuk keperluan pribadi.

Protokol-protokol *call setup* yang banyak digunakan pada komunikasi *real-time* seperti VoIP, *videoconference*, maupun *videophone* antara lain : H.323, MGCP (*Media Gateway Control Protocol*) dan SIP (*Session Initiation Protocol*). Aplikasi *videophone* yang dijalankan pada jaringan IP (*Internet Protocol*) merujuk pada rekomendasi ITU-T H.323, yaitu mengenai Sistem Komunikasi Multimedia Berbasis Paket (*Packet-Based Multimedia Communication System*).



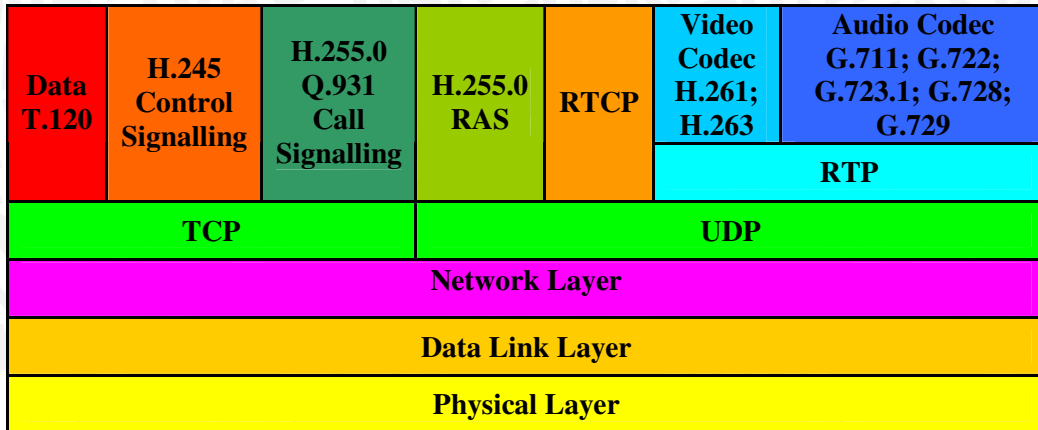
Gambar 2.8 Konfigurasi *videophone*
Sumber : Subchi, 2005 : 4

2.2.2 Standar Protokol H.323

Protokol H.323 merupakan rekomendasi dari *International Telecommunication Union* (ITU-T) digunakan pada jaringan berbasis paket seperti Ethernet, TCP/UDP/IP, ATM dan Frame Relay, dimana menggunakan *Internet Protocol* (IP) untuk *inter-network conferencing*

Pada aplikasi *videophone*, UDP (*User Datagram Protokol*) dan RTP (*Real-time Transport Protokol*) merupakan protokol yang dapat memberikan

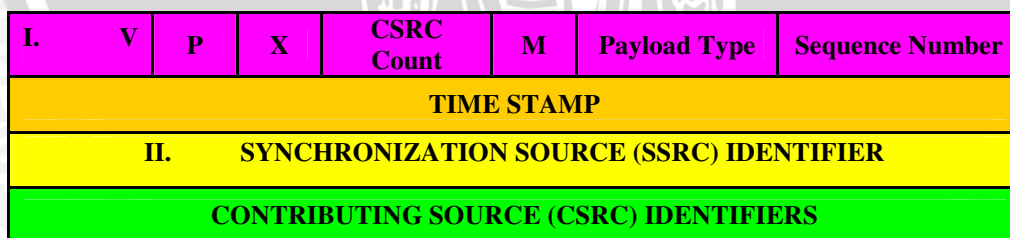
layanan *real-time* dimana tidak terjadi pengecekan kesalahan (*error correction*), sehingga *delay* yang terlalu besar dapat dihindari untuk aplikasi ini.



Gambar 2.9 H.323 protocol stack
Sumber : www.cis.ohio-state.edu

2.2.2.1 RTP (*Real-time Transport Protocol*)

RTP memberikan mekanisme transport yang memungkinkan terjadinya sinkronisasi *multimedia data stream* dari berbagai aplikasi misalnya *video* dan *voice*. Apabila *user* bekerja dengan aplikasi *video* dan *voice*, maka akan terdapat dua data *stream* yaitu *video* dan *voice*, dengan RTP maka paket yang berbeda itu akan dapat ditandai sesuai dengan *payload* masing-masing aplikasi sehingga pada *user* tujuan paket data tersebut dapat disinkronisasi. Format *header* dari sebuah paket RTP dapat dilihat pada Gambar 2.10.



Gambar 2.10 Format *header* RTP
Sumber : Subchi, 2005 : 5

- **P, Padding** bit. Jika bit ini diset, maka akan terdapat satu atau lebih oktet di akhir paket yang bukan merupakan bagian dari *payload*. Bit ini berfungsi untuk algoritma enkripsi.
- **V, 2-bit RTP version.**

- **X, Extension** bit. Jika bit ini diset, maka akan terdapat satu *extension header* yang mengikuti *fixed header*.
- **CSRC count**, 4-bit *count*, berisi nomor identitas dari *contributing source* yang mengikuti *fixed header*.
- **M, Marker bit**, digunakan untuk tujuan khusus.
- **Payload type**, 7-bit tipe *payload*, menspesifikasi format *payload* pada paket RTP.
- 16-bit **Sequence number**, setiap paket RTP mendapat sebuah nomor *sequence* yang digunakan pada bagian penerima untuk membentuk paket *sequence* dan mendeteksi paket rusak.
- **Timestamp**, berisi sebuah nilai yang merepresentasikan waktu dimana data dalam paket RTP disampling. Inisialisasi nilai ini secara acak.
- **SSRC identifier, Synchronization Source (SSRC)** merupakan sebuah identifikasi acak pada pengirim dari sebuah paket RTP, berupa nomor 32 bit. Hal ini berguna supaya penerima dapat mensinkronisasi kembali paket dari pengirim, dan setiap sesi memiliki identitas yang berbeda.
- **CSRC list**, berisi tentang *contributing sources* untuk *payload* dari paket terdahulu. *Field* ini digunakan apabila sebuah *mixer* telah mengkombinasi beberapa *stream* paket RTP dengan tipe *payload* yang sama (misalkan audio) dari pengirim yang berbeda. Hal ini berguna supaya penerima tetap menerima SSRC dari pengirim walaupun paket RTP sudah dimodifikasi oleh *mixer*.

Fungsi utama protokol RTP adalah :

- **Payload type identification**, *payload* dari sebuah paket RTP dapat berupa format *video* maupun *audio*. Aplikasi yang sedang berjalan menuliskan kode dari *payload* pada *header* RTP supaya aplikasi penerima mengetahui format *video* maupun *audio* yang harus dikodekan.
- **Sequence numbering**, *sequence numbering* digunakan pada RTP untuk membentuk kembali paket RTP pada penerima dan mendeteksi paket rusak/hilang. Selain itu juga berfungsi untuk mengetahui urutan asli dari suatu paket, misalnya pada saat mendekodekan *video*, supaya tidak harus

selalu mendekodekan secara berurutan. Nilai dari nomor ini adalah acak/tak dapat diperkirakan.

- *Time stamping, time stamp* berfungsi untuk mensinkronisasi paket dari sumber yang berbeda.

2.2.2.2 UDP (*User Datagram Protocol*)

UDP merupakan protokol *transport* yang sederhana. Berbeda dengan TCP yang *connection oriented*, UDP bersifat *connectionless*. Dalam UDP tidak ada *sequencing* (pengurutan kembali) paket yang datang, *acknowledgment* terhadap paket yang datang, atau retransmisi jika paket mengalami masalah di tengah jalan. Kemiripan UDP dengan TCP ada pada penggunaan *port number*. Sebagaimana digunakan pada TCP, UDP menggunakan *port number* ini untuk membedakan pengiriman datagram ke beberapa aplikasi berbeda yang terletak pada komputer yang sama. Karena sifatnya yang *connectionless* dan *unreliable*, UDP umumnya dipakai untuk transfer data yang memerlukan kecepatan tetapi kurang peka terhadap kesalahan, seperti transfer suara dan video. UDP ini bersifat *broadcasting* atau *multicasting*. Pengiriman datagram ke banyak klien sekaligus akan efisien jika prosesnya menggunakan metode *connectionless*.

Source Port	Destination Port
Datagram Length	Checksum
Application Data	

Gambar 2.11 Format datagram UDP
Sumber : Purbo, 2001:56

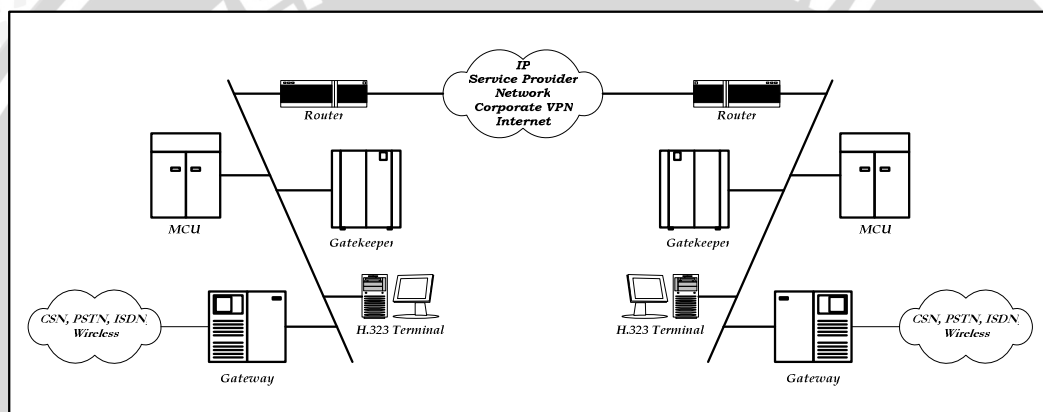
Pada Gambar 2.11 ditunjukkan format dari datagram UDP. *Source* dan *destination port* memiliki fungsi yang sama seperti pada TCP. *Datagram length* berisi panjang datagram, sedangkan *checksum* berisi angka hasil perhitungan matematis yang digunakan untuk memeriksa kesalahan data.

- *Source port*, menandakan *port* dimana proses pengiriman terjadi, sehingga apabila ada proses balasan akan menuju *port source*.
- *Destination port*, menandai *port* tujuan yang akan diproses yang merupakan *destination port*.

- *Length*, merupakan panjang (*byte*) dari *datagram* ini termasuk *header*-nya.
- *Checksum*, merupakan *optional field* yang terdiri dari 16 bit, yang berfungsi untuk mengecek *header* UDP dan data UDP.

2.2.3 Komponen H.323

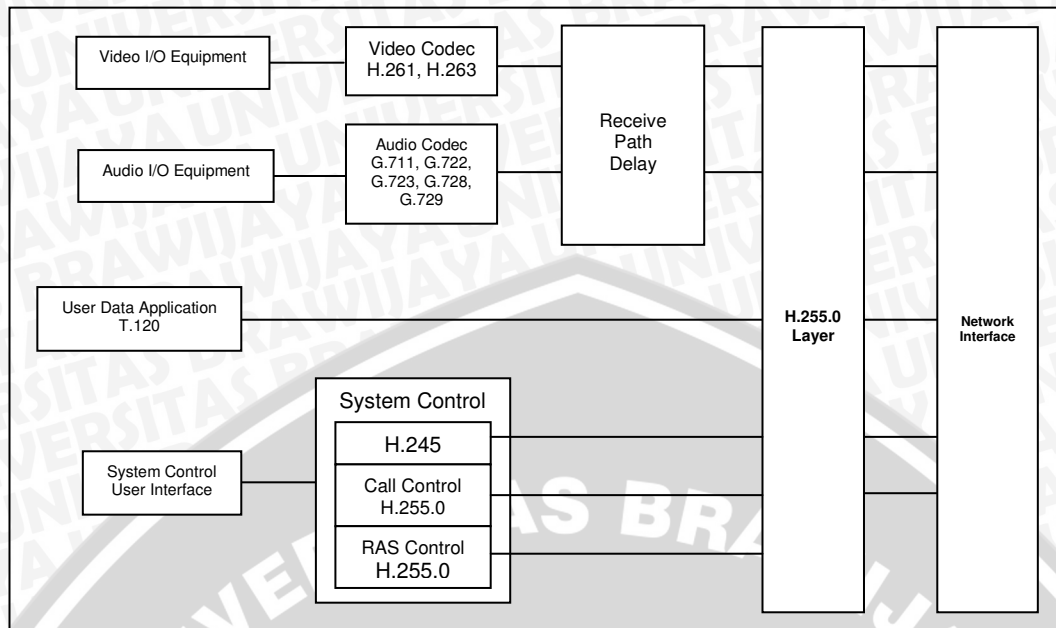
Rekomendasi ITU-T H.323 mendefinisikan komponen-komponen, prosedur-prosedur dan protokol-protokol yang penting untuk penyediaan komunikasi *audio visual*. Komponen-komponen H.323 terdiri dari 4 bagian, yaitu: terminal, *Gateway*, *Gatekeeper*, *Multipoint Control Unit* (MCU) [<http://www.cis.ohio-state.edu>].



Gambar 2.12 – Komponen H.323
 Sumber : www.cis.ohio-state.edu

2.2.3.1 Terminal

Gambar 2.13 menunjukkan H.323 *terminal equipment*. Elemen-elemen terminal rekomendasi H.323 adalah sebagai berikut :



Gambar 2.13 H.323 terminal equipment
 Sumber : ITU-T H.323, 2000 : 17

- *Video codec* (H.261, H.263), mengkodekan *video* dari sumber *video* (seperti kamera) untuk ditransmisikan dan mendekodekan kode *video* yang diterima yang dikeluarkan melalui *video display*.
- *Audio codec* (G.711, G.755, G.723, G.728, G.729), mengkodekan sinyal *audio* dari mikrofon untuk ditransmisikan dan mendekodekan kode *audio* yang diterima untuk dikeluarkan melalui *loudspeaker*.
- *Data channel* (T.120), mendukung aplikasi telematik seperti *image transfer*, pertukaran file, akses database, konferensi audiografik.
- *System Control Unit* (H.245, H.225), menyediakan pensinyalan pemakaian kanal untuk operasi terminal H.323 yang sempurna.
- *Media transmission*, mentransmisikan dan menerima informasi *audio*, *video*, data, aliran kontrol ke-dan-dari *interface* jaringan.
- *Network interface* atau interface jaringan, menyediakan *interface* ke jaringan yang berbasis paket.

2.2.3.2 Gateway

H.323 gateway merupakan komponen penghubung antar H.323 terminal dengan dalam sebuah jaringan IP, antar terminal antar jaringan IP dan antar H.323 gateway, serta menyediakan *real-time* dan komunikasi dua arah. H.323 gateway

juga menerjemahkan format transmisi *audio*, *video* dan data sesuai dengan protokol yang dipakai jika terdapat interkoneksi antara terminal H.323 dengan *Switched Circuit Network* (SCN). *Gateway* tidak diperlukan jika tidak ada interkoneksi dengan SCN.

2.2.3.3 Gatekeeper

H.323 *gatekeeper* merupakan komponen yang berfungsi sebagai *central point* untuk semua panggilan dalam satu zona dan menyediakan registrasi *endpoint*. Fungsi-fungsi lain dari *gatekeeper*, yaitu :

- *Address translation*, membentuk suatu *database* untuk translasi antara nomor telepon internasional dengan alamat jaringan.
- *Admission and access control of endpoints*, berfungsi untuk mengontrol *bandwidth* yang tersedia, membatasi jumlah panggilan, dan mendaftarkan *endpoint*.
- *Bandwidth management*, membantu *network administrator* mengatur *bandwidth* dengan cara membatasi jumlah panggilan, dan dengan cara membatasi terminal untuk melakukan panggilan pada waktu tertentu.
- *Routing capability*, *gatekeeper* dapat merutekan semua panggilan, sehingga akan mempermudah proses perhitungan biaya dan keamanan. Selain itu dapat merutekan panggilan ke *gateway* yang mempunyai *bandwidth* berlebih dan juga bisa menyediakan servis tambahan seperti *mobile addressing*, *call forwarding*, dan *voice mail diversion*.

2.2.3.4 Multipoint Control Unit (MCU)

MCU adalah *endpoint* yang dapat mendukung konferensi *multipoint* dan minimum terdiri dari satu *multipoint controller* (MC), dan satu atau lebih *multipoint processor* (MP). *Multipoint controller* berfungsi sebagai sentral dari panggilan *multipoint*, dimana panggilan dan *control signalling* dirutekan melalui MC sehingga kemampuan *endpoint* bisa diketahui dan parameter komunikasi bisa diatur. MC juga dapat digunakan dalam sebuah panggilan *point-to-point* yang bisa dikembangkan menjadi *multipoint conference*. Selain itu MC juga dapat mengatur apakah *audio* dan *video stream* dikirim secara *unicast* atau *multicast* tergantung dari kemampuan jaringan dan topologi dari *multipoint conference*.

MP menerima berkas *audio*, *video*, dan/atau data dan mendistribusikan berkas tersebut ke *endpoint* yang berpartisipasi dalam *multipoint conference*.

2.2.4 Mekanisme H.323 Dalam Melakukan *Call Setup*

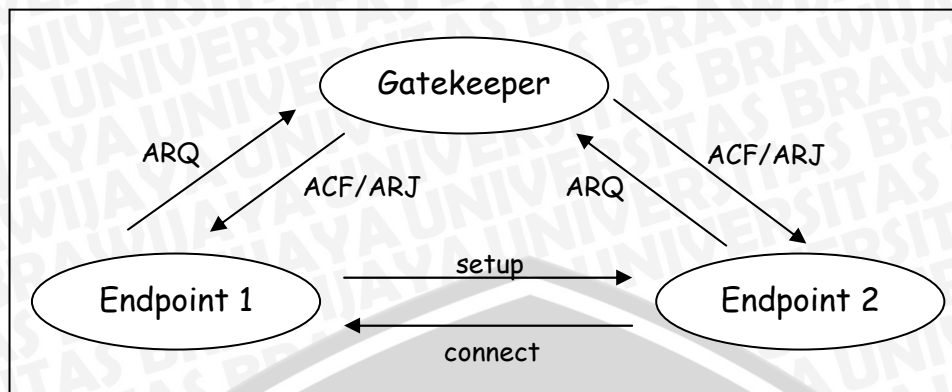
Proses pembentukan hubungan atau *call setup* pada aplikasi *videophone* yang menggunakan standar H.323 melibatkan terminal/*endpoint* dan *gatekeeper*. Proses ini dapat dilakukan setelah *endpoint* teregistrasi di *gatekeeper*.

2.2.4.1 *Gatekeeper Discovery Dan Registrasi*

Gatekeeper discovery adalah proses yang dilakukan *endpoint* untuk mengidentifikasi ke *gatekeeper* mana *endpoint* harus teregistrasi. Proses registrasi adalah proses yang dilakukan *endpoint* untuk mendaftarkan diri ke *gatekeeper* dan memberitahukan *gatekeeper* alamat IP dan alias *endpoint* tersebut. Alias adalah alamat yang digunakan untuk mewakili alamat IP *endpoint*. Proses ini dapat dilakukan dalam dua cara, yaitu manual dan otomatis. Pada metode manual tiap *endpoint* dikonfigurasi dengan alamat IP *gatekeeper* yang akan digunakan, sehingga *endpoint* dapat langsung menghubungi *gatekeeper* untuk melakukan registrasi. Pada metode otomatis alamat *transport gatekeeper* diperoleh *endpoint* dengan cara mengirim pesan (*request*) *multicast* dan *gatekeeper* akan mengirim pesan jawaban (*confirm*) yang berisi alamat *transport gatekeeper* tersebut.

2.2.4.2 Proses Pembentukan Hubungan/*Call Setup*

Proses pembentukan hubungan/*call setup* antar *endpoint* ditunjukkan pada gambar 2.14 dibawah ini. Sebagai contoh, pada gambar tersebut dimisalkan *endpoint* 1 ingin menghubungi *endpoint* 2 dan kedua *endpoint* teregistrasi pada *gatekeeper* yang sama. Proses tersebut, dijelaskan sebagai berikut :

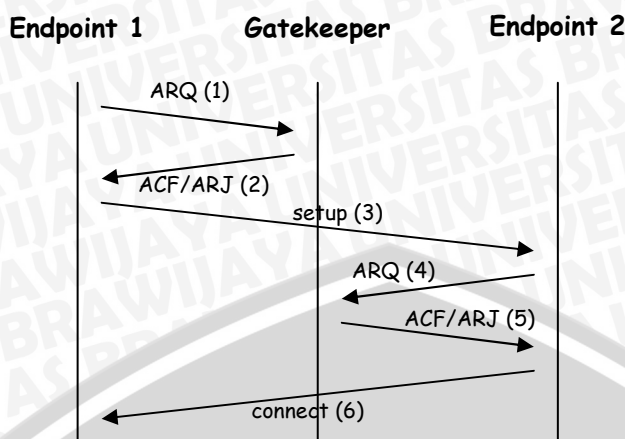


Gambar 2.14 Proses pembentukan hubungan aplikasi videophone menggunakan standar H.323 (menggunakan *direct endpoint call signalling*)

Sumber : Davidson, 2000 : 240

1. Ketika pengguna di *endpoint 1* menekan nomor telepon *endpoint 2*, *endpoint 1* akan mengirim pesan *Admission Request* (ARQ) ke *gatekeeper*. Pesan ARQ dikirim untuk menanyakan ke *gatekeeper* apakah *endpoint* tersebut boleh melakukan panggilan.
2. *Gatekeeper* memeriksa daftar registrasi dan menemukan *endpoint 2* teregistrasi dengan nomor yang dikirim *endpoint 1* tersebut. Selanjutnya *gatekeeper* mengirimkan pesan *Admission Confirm* (ACF) ke *endpoint 1* beserta alamat IP *endpoint 2*. ACF dikirim untuk memberitahukan *endpoint 1* bahwa *endpoint 1* dapat memulai panggilan ke *endpoint 2*.
3. Setelah menerima ACF, *endpoint 1* telah mengetahui alamat IP *endpoint 2*. Selanjutnya *endpoint 1* akan mengirim pesan *setup* ke *endpoint 2* untuk melakukan panggilan.
4. Setelah menerima pesan *setup*, *endpoint 2* mengirim pesan ARQ ke *gatekeeper* untuk menanyakan apakah *endpoint 2* diijinkan untuk menjawab panggilan *endpoint 1*.
5. Jika *gatekeeper* mengizinkan *endpoint 2* untuk menerima panggilan *endpoint 1*, *gatekeeper* mengirim pesan ACF beserta alamat IP *endpoint 1* ke *endpoint 2*.
6. Setelah pengguna *endpoint 2* mengangkat *handset*, *endpoint 2* akan mengirim pesan *connect* ke *endpoint 1*. Hubungan dapat dimulai.

Proses *call setup* diatas dapat digambarkan dengan skema seperti pada gambar 2..15 dibawah ini.



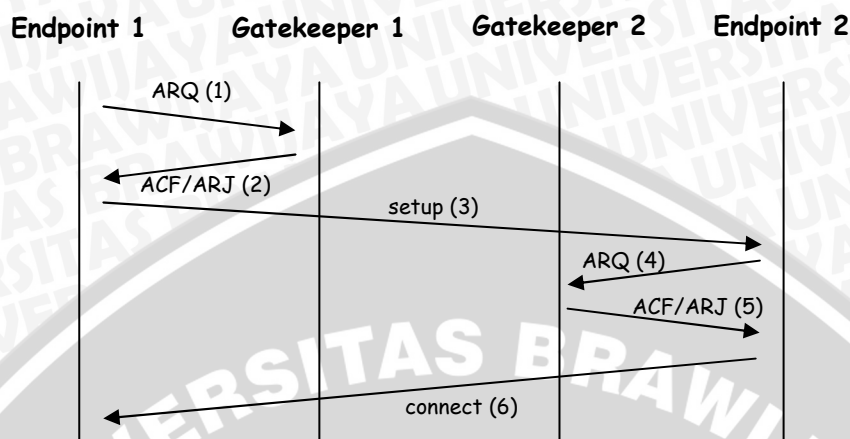
Gambar 2.15 Skema proses call setup antara 2 endpoint yang teregistrasi pada gatekeeper yang sama (menggunakan *direct endpoint call signalling*)
 Sumber : ITU-T H.323, 2000 : 82

Sedangkan skema proses *call setup* antara dua *endpoint* yang teregistrasi pada *gatekeeper* yang berbeda ditunjukkan dalam gambar 2.16 (menggunakan *Direct Endpoint Call Signalling*).

Proses tersebut dijelaskan sebagai berikut :

1. Ketika pengguna di *endpoint 1* menekan nomor telepon *endpoint 2*, *endpoint 1* akan mengirim pesan *Admission Request* (ARQ) ke *gatekeeper*. Pesan ARQ dikirim untuk menanyakan ke *gatekeeper* apakah *endpoint* tersebut boleh melakukan panggilan.
2. *Gatekeeper 1* mengirim pesan *Admission Confirm* (ACF) ke *endpoint 1* beserta alamat IP *endpoint 2*, jika *gatekeeper 1* dihubungkan dengan *gatekeeper 2*.
3. Setelah menerima ACF, *endpoint 1* telah mengetahui alamat IP *endpoint 2*. Selanjutnya *endpoint 1* akan mengirim pesan *setup* ke *endpoint 2* untuk melakukan panggilan.
4. Setelah menerima pesan *setup*, *endpoint 2* mengirimkan pesan ARQ ke *gatekeeper 2* untuk menanyakan apakah *endpoint 2* diijinkan untuk menjawab panggilan *endpoint 1*.
5. Jika *gatekeeper 2* mengizinkan *endpoint 2* untuk menerima panggilan *endpoint 1*, *gatekeeper 2* mengirimkan pesan ACF beserta alamat IP *endpoint 1* ke *endpoint 2*.

- Setelah pengguna *endpoint 2* mengangkat *handset*, *endpoint 2* akan mengirim pesan *connect* ke *endpoint 1*, dan hubungan dapat dilakukan.



Gambar 2.16 Skema proses call setup antara 2 endpoint yang teregistrasi pada gatekeeper yang berbeda (menggunakan *direct endpoint call signalling*)
 Sumber : ITU-T H.323, 2000 : 87

2.2.5 CODEC (COder/DECoder)

Dalam aplikasi *videophone* membutuhkan beberapa perangkat seperti kamera *video*, layar *video (display video)*, sistem pemrosesan suara dan pemrosesan gambar. *CODEC* merupakan bagian dari *hardware* maupun *software* yang memproses suatu sinyal (*video* maupun *audio*) dan mengkonversinya kedalam format tertentu untuk dikirimkan dengan protokol tertentu dan pada sisi penerima juga harus menggunakan protokol yang sama.

Pada aplikasi multimedia baik itu untuk *voice*, *video* maupun data besarnya *bandwidth* tergantung dari *CODEC* yang digunakan. Teknik *CODEC* yang direkomendasikan ITU-T untuk aplikasi *videophone* dapat dilihat pada tabel 2.1

Tabel 2.1 Audio dan video CODEC

Audio CODEC	Bit Rate	Delay CODEC
G.711	64 kbps	0.75 ms
G.726	32 kbps	1 ms
G.723.1	5,3/6,3 kbps	30 ms
G.728	16 kbps	3-5 ms
G.729	8 kbps	10 ms

Video		
CODEC		
H.261	P x 64 kbps	20-150 ms
H.263	10-384 kbps	20-150 ms

Sumber : www.cis.ohio-state.edu

Jenis *CODEC* yang direkomendasikan oleh ITU-T untuk aplikasi *videophone* menggunakan standar H.263 untuk *video* dan G.723.1 untuk *audio*, dimana *bandwidth* yang digunakan tidak terlalu besar. Semakin besar *bandwidth* semakin besar pula kebutuhan tempat penyimpanan dan media transmisi sinyal *video* dan *audio* tersebut, membuat biaya pelaksanaan menjadi mahal.

2.2.6 Format Gambar

Banyaknya informasi gambar tergantung dari tiga elemen utama yaitu ukuran gambar dan jumlah garis yang digunakan untuk membangkitkan *image*, kecepatan pengulangan pada gambar dan jumlah resolusi yang diperlukan untuk *image*. Standar TV yang dipakai di Amerika dan Jepang yaitu standar NTSC memiliki 262.5 garis, 525 elemen penggambar (*pel*) dan pelarikan (*scanning*) tiap detiknya adalah 30 *frame*, dan tiap *pixel* dikodekan oleh 8 bit sehingga besar kecepatan datanya sebesar [*The widening scop of video conferencing* : hal. 32] :

$$\begin{aligned}
 \text{Kecepatan bit datanya} &= \text{jumlah } \textit{pixel} \text{ per garis} \times \text{jumlah garis per} \\
 &\quad \textit{frame} \times \text{jumlah } \textit{pixel} \text{ yang dikodekan} \times \\
 &\quad \text{resolusi kecepatan bit} \\
 &= 525 \textit{ pixel} \text{ per garis} \times 262.5 \text{ garis per } \textit{frame} \times \\
 &\quad 30 \textit{ frame} \text{ per detik} \times 8 \text{ bit per } \textit{pixel}. \\
 &= 33075000 \text{ bit/detik atau } 33 \text{ Mbit/detik}.
 \end{aligned}$$

Jadi dengan sistem NTSC tanpa pemampatan memerlukan kecepatan bit 33 Mbps.

Sedangkan untuk sistem PAL yang dipakai di Eropa dan juga di Indonesia struktur standar sistemnya adalah terdiri dari 625 *pixel* per garis, 312.5 garis per *frame* dan pelarikannya 25 Hz, dan tiap *pixel*nya dikodekan 8 bit, kecepatannya bitnya dapat dihitung seperti dibawah ini :

$$\begin{aligned}
 \text{Kecepatan bit datanya} &= 625 \textit{ pixel} \text{ per garis} \times 312.5 \text{ garis per } \textit{frame} \times \\
 &\quad 25 \textit{ frame} \text{ per detik} \times 8 \text{ bit per } \textit{pixel}.
 \end{aligned}$$

= 39062500 bit/detik atau 39 Mbit/detik.

Jadi dengan sistem PAL ini akan menghasilkan kecepatan bit data sekitar 39 Mbps.

Gambar berwarna yang mengandung 3 sinyal warna pokok *red*, *green*, *blue* (*RGB*) dengan ukuran yang sama, memerlukan jumlah data 3 kali lipat yaitu 39 x 39 Mbps. Jadi kecepatan datanya 117 Mbps.

Pada TV *broadcast*, *bandwidth* sebesar itu merupakan suatu standar dengan menggunakan 30 *frame* per detik untuk NTSC dan 25 *frame* per detik untuk PAL. Format gambar yang biasa dipakai untuk aplikasi *videophone* ada dua, yaitu CIF (*Common Intermediate Format*) dan QCIF (*Quarter Common Intermediate Format*). CIF memiliki kualitas setengah dari kualitas NTSC dimana kualitas itu sama dengan kualitas VCR (*Video Cassette Recorder*). Format ini direkomendasikan untuk aplikasi *multi videoconference*, karena membutuhkan *bandwidth* yang cukup besar (sekitar 384 kbps). QCIF merupakan format alternatif untuk aplikasi *video*, dimana resolusinya adalah setengah dari resolusi CIF, sehingga kualitasnya dibawah dari CIF. Sehingga *bandwidth* dengan 15 *frame* per detik maupun 10 *frame* perdetik masih dianggap memenuhi syarat untuk aplikasi *videophone*, walaupun tidak sebgus NTSC maupun PAL.

Tabel 2.2 Konsumsi *bandwidth* aplikasi *videophone*

Nominal Bandwidth	Size	Frame Rate	IP Overhead (%)	Actual Bandwidth
128 kbps	QCIF	5 ms	9,1 %	140 kbps
174 kbps	CIF	8-10 ms	7,9 %	189 kbps
384 kbps	CIF	15 ms	6,0 %	408 kbps
768 kbps	CIF	30 ms	5,2 %	810 kbps

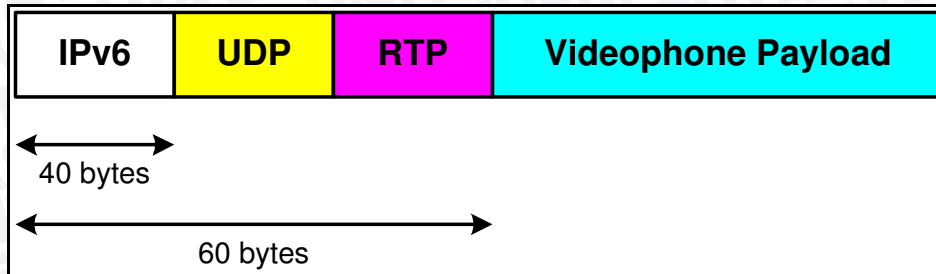
Sumber : PictureTel, 2000:9

2.2.7 Paket Data Aplikasi *Videophone*

Paket data yang dihasilkan merupakan *payload* dari aplikasi ditambah *header* RTP, *header* UDP, dan *header* IP. Karena pada skripsi ini IP yang digunakan adalah IP versi 6 sehingga *header* yang digunakan adalah *header* IPv6 sebesar 40 byte. Pada aplikasi *videophone* bentuk paket yang dihasilkan dapat dilihat pada gambar 2.17.



Besar *header* tiap paket adalah *header* IPv6 40 byte, *header* UDP 8 byte, *header* RTP 12 byte. Sehingga total *header* adalah 60 byte.



Gambar 2.17 Paket data *videophone*

Sumber : Subchi, 2005 : 18

Pada aplikasi *videophone* paket yang ditransmisikan dibedakan atas paket *audio* dan paket *video*, dimana tiap paket tersebut mempunyai besar *payload* yang berbeda. Besar *payload* tiap paket *audio* dan *video* dapat dinyatakan dengan [http://www.erlang.com/bandwidth.html]:

$$PL_a = B_{CODECA} \times \text{frame rate} \quad (2.1)$$

$$PL_v = B_{CODECV} \times \text{frame rate} \quad (2.2)$$

dengan :

$$PL_a = \text{payload tiap paket audio (bit)}$$

$$PL_v = \text{payload tiap paket video (bit)}$$

$$B_{CODECV} = \text{bandwidth CODEC video (bps)}$$

$$B_{CODECA} = \text{bandwidth CODEC audio (bps)}$$

$$\text{Frame rate} = \text{waktu pembentukan tiap frame (detik)}$$

Sedangkan besar tiap paket dan jumlah paket yang dihasilkan dapat dinyatakan dengan [Cisco,2002 : 2] :

$$Pa\text{-size} = \text{header} + PL_a \quad (2.3)$$

$$Pa = B_{CODECA} / PL_a \quad (2.4)$$

$$Pvp\text{-size} = \text{header} + (PL_a + PL_v) \quad (2.5)$$

dimana :

$$Pa\text{-size} = \text{besar tiap paket audio (bit)}$$

$$Pa = \text{jumlah paket audio yang dihasilkan tiap detik (paket per detik)}$$

$$\text{header} = \text{total header tiap paket (60 byte/480 bit)}$$

$$Pvp\text{-size} = \text{ukuran paket aplikasi videophone (bit)}$$

Pada kenyataannya *bandwidth* aplikasi *videophone* adalah jumlah *bandwidth videophone (CODEC)* yang ditambah dengan *bandwidth overhead*. *Bandwidth overhead* merupakan *bandwidth* akibat adanya penambahan *header* pada setiap paket dari *videophone* yang besarnya sekitar 5 – 10% dari *bandwidth videophone* [Bartlett,John,2000 : 7].

Sehingga besarnya *bandwidth* aktual dari *videophone* dapat dinyatakan dengan [Cisco,2002 : 2] :

$$B_a = P_a\text{-size} \times P_a \quad (2.6)$$

$$B_v = P_v\text{-size} \times P_v \quad (2.7)$$

$$B_{vp} = B_v + B_a + \text{bandwith overhead} \quad (2.8)$$

dimana :

$$B_a = \text{bandwidth aktual audio (bps)}$$

$$B_v = \text{bandwith aktual video (bps)}$$

$$B_{vp} = \text{bandwidth aktual aplikasi videophone}$$

2.2.8 Delay End-to-end Aplikasi Videophone

Delay end-to-end aplikasi *videophone* merupakan jumlah *delay CODEC* aplikasi *videophone* dan *delay gatekeeper* ditambah dengan *delay* jaringan dimana aplikasi itu berjalan, khusus untuk *delay* jaringan akan dibahas pada bab selanjutnya. *Delay CODEC* aplikasi *videophone* terdiri dari *delay CODEC audio* dan *video*. Besarnya *delay CODEC* aplikasi *videophone* ini dapat dihitung dengan [Compaq, 1999: 6]:

$$t_{CODEC} = t_a + t_v \quad (2.9)$$

dimana :

$$t_{CODEC} = \text{delay CODEC aplikasi videophone (detik)}$$

$$t_a = \text{delay CODEC audio (detik)}$$

$$t_v = \text{delay CODEC video (detik)}$$

Delay gatekeeper dalam hal ini merupakan *delay* yang dialami paket selama berada pada sistem *gatekeeper* sebelum memasuki segmen jaringan berikutnya, termasuk waktu antrian dan waktu pelayanan mencakup pelayanan *gatekeeper* untuk mengatur dan mengontrol *bandwidth* dengan cara membatasi terminal untuk melakukan panggilan pada waktu tertentu, serta merutekan panggilan ke *gateway* yang mempunyai *bandwidth* berlebih untuk peningkatan

efisiensi jaringan. Waktu rata-rata setiap paket data dalam sistem *gatekeeper* dihitung dengan persamaan [Wirjana, 1999] :

$$t_{gk} = \frac{1}{\lambda_{gk} - \mu_{gk}} \quad (2.10)$$

dimana :

$$\begin{aligned} t_{gk} &= \text{delay gatekeeper (detik)} \\ \mu_{gk} &= \text{rata-rata pelayanan di gatekeeper (paket/detik)} \\ \lambda_{gk} &= \text{rata-rata kedatangan paket gatekeeper (paket/detik)} \\ \mu_{gk} &= \frac{P}{C} \times 8 \end{aligned} \quad (2.11)$$

dimana :

$$\begin{aligned} P &= \text{ukuran paket data di gatekeeper (bit)} \\ C &= \text{kecepatan standar gatekeeper (bps)} \\ T_{rx} &= \frac{P}{C} + \text{interpacketgap} \end{aligned} \quad (2.12)$$

Interpacket gap = 3 μ s (100 Mbps)

Delay end-to-end aplikasi *videophone* adalah 250 ms (*voice* sebagai prioritas) untuk jaringan IP. Sehingga Total *delay end-to-end* dapat dihitung dengan persamaan [Compaq, 1999:6]:

$$t_{tot} = t_{CODEC} + t_{gk} + t_{net} \quad (2.13)$$

dimana :

$$\begin{aligned} t_{tot} &= \text{delay end-to-end aplikasi videophone (detik)} \\ t_{net} &= \text{delay jaringan (detik)} \end{aligned}$$

2.2.9 Probabilitas *Packet Loss* Aplikasi *Videophone*

Probabilitas *packet loss* total aplikasi *videophone* pada suatu jaringan ditentukan berdasarkan pada probabilitas *packet loss* pada jaringan tersebut serta probabilitas *packet loss* aplikasi *videophone* yang berbasis protokol UDP/RTP/IP seperti pada persamaan [Pritchard,et.al.1993:339]:

$$\rho_{tot} = 1 - [(1 - \rho_{Network})(1 - \rho_{UDP/RTP/IP})] \quad (2.14)$$

dengan :

$$\rho_{tot} = \text{probabilitas packet loss total/end-to-end}$$

ρ_{Network} = probabilitas *packet loss* pada jaringan

$\rho_{\text{IP/UDP/RTP}}$ = probabilitas *packet loss* UDP/RTP/IP

Protokol UDP/RTP/IP sendiri tidak dapat memberikan jaminan bahwa paket akan dikirim semua sesuai dengan permintaan. *Packet loss* yang terjadi pada aplikasi *videophone* dapat dihitung dengan persamaan dibawah [Apostolopoulos, John : 2001] :

$$\rho_{\text{UDP/RTP/IP}} = P_{\text{size}} \cdot \rho_b \quad (2.15)$$

$$P_{\text{size}} = (H_{\text{UDP/RTP/IP}} + PL_{\text{VP}}) \quad (2.16)$$

dimana :

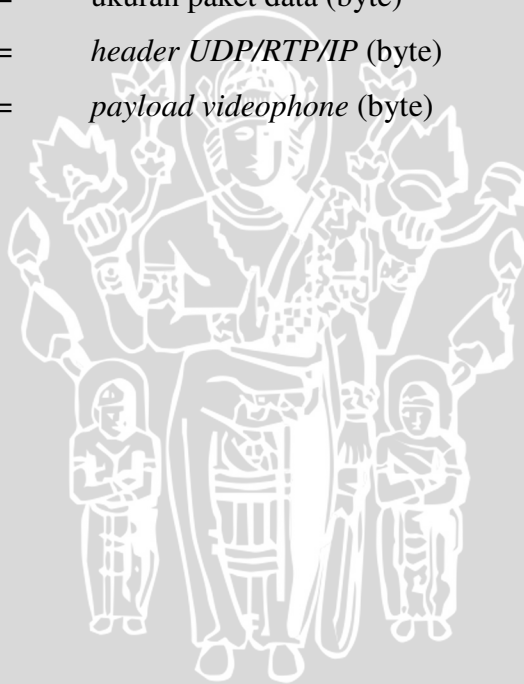
$\rho_{\text{UDP/RTP/IP}}$ = probabilitas *packet loss* aplikasi *videophone*

ρ_b = BER (10^{-7})

P_{size} = ukuran paket data (byte)

$H_{\text{UDP/RTP/IP}}$ = *header UDP/RTP/IP* (byte)

PL_{VP} = *payload videophone* (byte)



BAB III

PENERAPAN IPv6 UNTUK APLIKASI *VIDEOPHONE* PADA JARINGAN UMTS

3.1 Umum

Meningkatnya kebutuhan masyarakat akan sarana telekomunikasi yang *real time*, praktis, serta mampu mengatasi masalah ruang yang cakupannya sangat luas bahkan sulit terjangkau, merupakan faktor utama dalam kemajuan teknologi telekomunikasi saat ini. Atas dasar inilah yang mendorong pesatnya perkembangan dalam sistem komunikasi bergerak hingga saat ini.

Saat ini sistem *mobile wireless* sebagian besarnya masih menggunakan sistem generasi ke-2 (2G) ataupun 2.5G. Sistem ini masih memiliki kekurangan dalam hal kecepatan data dan layanan multimedia. Namun demikian, memasuki milenium ketiga ini telah distandarkan sistem komunikasi bergerak generasi ketiga (3G) oleh *European Telecommunication Standar Institute (ETSI)* yang dikenal dengan *Universal Mobile Telecommunications System (UMTS)*. Teknologi UMTS merupakan langkah maju dalam hal layanan dan kemampuan kepada pengguna sebagai jawaban atas semakin tingginya kapasitas dan kualitas layanan yang dibutuhkan seperti : tahan terhadap *noise*, mempunyai *Bit Error Rate (BER)* yang kecil, efisiensi penggunaan spektrum frekuensi, peningkatan kapasitas trafik serta akan lebih mengeksplorasi kemampuan multimedia.

Sebagai evolusi dari generasi sebelumnya, UMTS memberikan beberapa kelebihan. UMTS membuka kesempatan untuk menyediakan layanan *broadband multimedia yang mobile*. Layanan multimedia memudahkan pengiriman informasi audio, visual dan berbasis-teks yang beragam. Pemfokusan pada teknik *packet switching* dan *internet protocol* memberikan kapasitas dan efisiensi yang lebih tinggi. UMTS memberikan kecepatan data sampai 2 Mbit/s pada frekuensi 2 GHz.

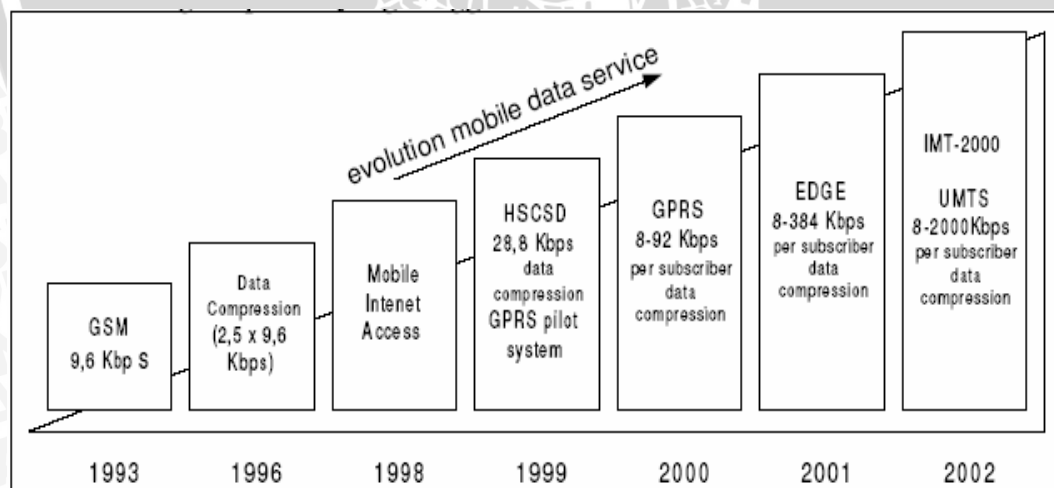
Kehadiran sistem telekomunikasi generasi ketiga (3G) dengan kemampuannya untuk memproses aplikasi multimedia secara *real-time* juga adanya *bandwith* yang besar akan mendorong kemajuan yang pesat dalam dunia *internet*. Dengan sistem baru ini layanan multimedia yang berupa telepon *mobile*, akses *internet* dan *audio video atau videophone* dapat dilayani secara bersama.

3.2 Konsep Dasar UMTS

Agar mampu memenuhi tuntutan akan nilai tambah pada layanan-layanan generasi ketiga, UMTS harus memenuhi beberapa persyaratan diantaranya adalah (UMTS Forum):

- Integrasi layanan residen, perkantoran, dan seluler dalam satu sistem tunggal yang berbasis pada satu perangkat pengguna.
- Keamanan komunikasi yang terjamin.
- Jaminan kualitas layanan yang tinggi, minimal sama dengan jaringan yang sudah ada.
- *Handset* yang *portable* dengan batere yang panjang.
- Kemampuan menyediakan layanan multimedia yang bervariasi.
- Memiliki kapasitas dan efisiensi penggunaan spektrum yang tinggi untuk mendukung penetrasi pasar.
- Cakupan radio secara global.
- Memungkinkan mengakses ke berbagai layanan dengan satu tipe terminal.

Kriteria-kriteria tersebut menunjukkan bahwa sistem UMTS didesain untuk *multi service* (berbagai layanan), *multi operator* (operator yang beragam) dan *multi operation environment* (lingkungan operasi yang berbeda-beda).



Gambar 3.1 Skema perkembangan layanan sistem komunikasi seluler

Sumber : Forum UMTS. Report No. 1, 1999

Jika dalam sistem GSM pada umumnya masih menggunakan *circuit switch* untuk transmisi, maka dalam UMTS menggunakan dua macam *switch mode* :

1. *Circuit Switch (CS)* yaitu suatu *mode* transfer dimana fungsi *switching* dan transmisi dikirim dengan alokasi tetap.
2. *Packet Switch (PS)* yaitu suatu *mode* transfer dimana fungsi *switching* dan transmisi dikirim dalam bentuk paket, sehingga memungkinkan untuk *sharing* dengan jaringan lainnya.

Kelebihan yang diberikan oleh sistem UMTS ini antara lain adalah memberikan akses untuk enam macam layanan. Masing-masing layanan tersebut mempunyai spesifikasi seperti dalam tabel 3.1.

Tabel 3.1 Macam-macam layanan UMTS dan karakteristiknya

Jenis Layanan	Bit Rate max (kbps)	Coding Factor	Asym. Factor	Service Bandwidth (kbit/s)	Switch Mode
High Interactive MM	128	2	1/1	256/256	Circuit
High MM	2000(384)	2	0,005/1	15/3200	Packet
Medium MM	384	2	0,026/1	15/572	Packet
Switched Data	14	3	1/1	43/43	Circuit
Simple Messaging	14	2	1/1	22/22	Packet
Speech	16	1,75	1/1	28/28	Circuit

Sumber : UMTS Forum

3.3 Spektrum Frekuensi

Spektrum frekuensi yang ditetapkan untuk UMTS berdasarkan WARC-92 (*World Administrative Radio Conference*) terletak pada pita frekuensi 1885-2025 MHz dan 2110-2200 MHz, dengan pembagian sebagai berikut :



1. *Frekuensi Division Duplex* (FDD, WCDMA), pasangan (*paired band*) 1920-1980 MHz (*uplink*) dan 2110-2170 MHz (*downlink*) dengan lebar kanal 5 MHz dan spasi 200 KHz.
2. *Time Division Duplex* (TDD, TD/CDMA), 1900-1920 MHz, dan 2010-2025 MHz bukan *band* yang berpasangan (*unpaired band*), dengan lebar kanal 5 MHz dan spasi 200 KHz.
3. *Mobille satellite*, 1980-2010 MHz (*uplink*) dan 2170-2200 MHz (*downlink*)

Alasan utama UMTS menggunakan dua metode akses jamak adalah untuk meningkatkan kualitas layanan ketika terjadi kepadatan trafik yang tinggi di area *pico cell* dan *micro cell* serta pengaksesan data asimetris (tidak sama) antara arah transmisi *uplink* dan *downlink*, seperti layanan internet, audio-video, multimedia, dan *videoconference*.

Mode FDD dioperasikan untuk melayani *macro cell* dan *micro cell*, sedangkan *mode* TDD dioperasikan untuk *pico cell*. *Mode* TDD tidak mengijinkan *delay* propagasi yang besar antara *mobile station* dan *base station* yang akan menyebabkan suatu benturan antara *time slot* saat *transmit* dan *receive*. Oleh karena itu *mode* ini hanya dapat digunakan di dalam lingkungan yang *delay* propagasinya kecil (*pico cell*). Namun *mode* TDD mempunyai keuntungan pada sebuah asimetris yang besar pada transfer data antara *uplink* dan *downlink*. Beberapa aplikasi *internet* ditandai oleh asimetris data yang besar dimana transfer data yang diterima lebih besar (*downlink*) dari pada yang ditransmisikan (*uplink*) (Thomas Nuebauer, 1998).

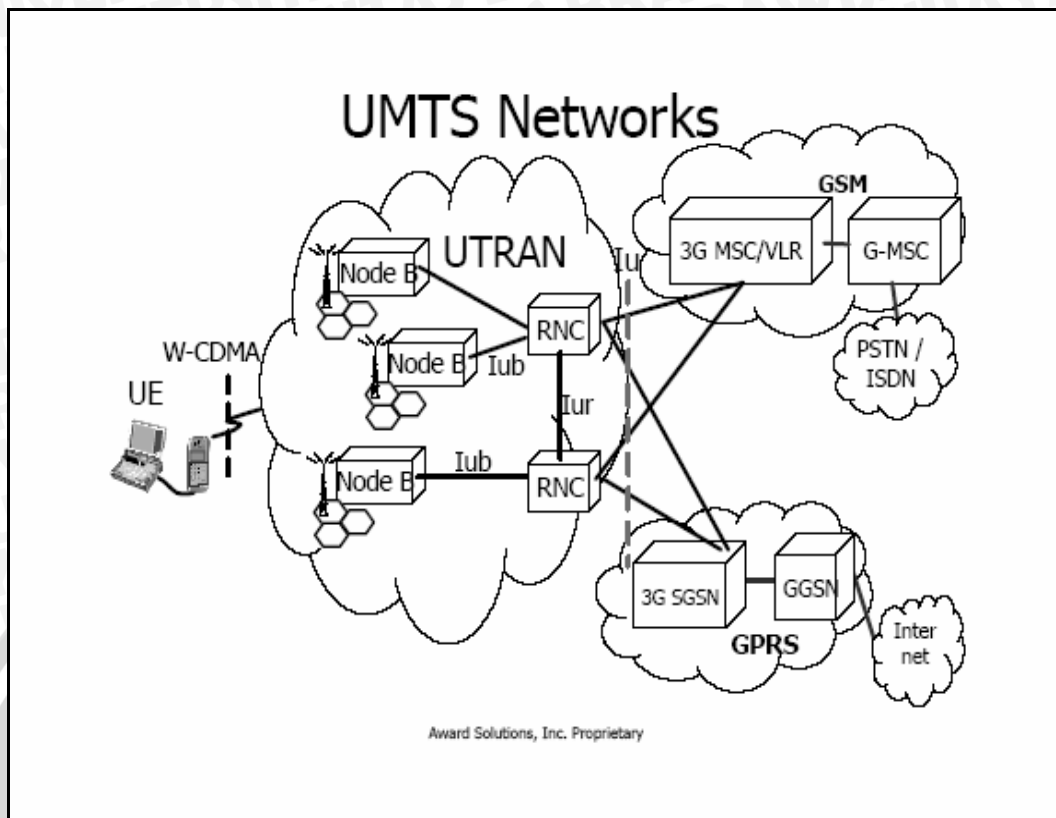
Mode FDD menggunakan metode akses yang berbeda dengan *mode* TDD. Untuk *mode* FDD, *multiple access* yang digunakan adalah *Wide band Code Division Multiple Access* (W-CDMA) sedangkan untuk *mode* TDD menggunakan *Time Division CDMA* (TD-CDMA). W-CDMA adalah sebuah variasi pada prinsip CDMA dengan *bandwidth* transmisi yang lebih lebar dan menggunakan *band* yang berpasangan (FDD). Sedangkan TD-CDMA sama seperti W-CDMA yang menggunakan prinsip CDMA tetapi digunakan pada *band* frekuensi yang tidak berpasangan (TDD).

3.4 Konfigurasi UMTS

Sistem UMTS terdiri dari sejumlah unsur-unsur jaringan yang masing-masing telah ditetapkan secara fungsional. Secara fungsional unsur-unsur jaringan dikelompokkan ke dalam *Radio Access Network* (RAN, UMTS *Terrestrial RAN* = UTRAN) yang menangani semua radio yang berhubungan dengan fungsinya, dan *Core Network* (CN) yang bertanggung jawab untuk *switching* dan *routing* panggilan ke jaringan *eksternal*. Untuk melengkapi sistem ini, maka ditetapkan *User Equipment* (UE) yang menghubungkan *user* dan *radio interface*. Gambar 3.2 menunjukkan konfigurasi umum jaringan UMTS yang terdiri atas *Core Network* (CN), *UMTS Terrestrial Radio Access Network* (UTRAN), dan *User Equipment* (UE).

Beberapa *interface* yang digunakan ketika setiap elemen pada UMTS akan saling berhubungan :

- *Uu interface*, menghubungkan UE (*User Equipment*) dengan *Node B*
- *Iu interface*, menghubungkan RNC dengan jaringan GSM dan CN (MSC/VLR atau SGSN).
 - *Iu-CS* untuk *circuit-switched data*
 - *Iu-PS* untuk *packet-switched data*
- *Iub interface*, menghubungkan RNC dengan *Node B*
- *Iur interface*, menghubungkan RNC dengan RNC



Gambar 3.2 Konfigurasi jaringan UMTS

Sumber : www.awardsolution.com, 2001

3.4.1 Core Network (CN)

Core Network merupakan entitas fisik yang memberikan layanan telekomunikasi dan fitur-fiturnya pada jaringan. Elemen-elemen penting yang ada didalamnya adalah :

1. *Home Location Register (HLR)*
Merupakan perangkat *database* yang diletakkan dalam sistem pengguna yang menyimpan *copy master* dari profil layanan pengguna.
2. *Mobile Service Switching Centre /Visitor Location Register (MSC/VLR)*
Adalah *switch (MSC)* dan *database (VLR)* yang melayani UE di lokasi kejadian untuk layanan *circuit switch (CS)*.
3. *Gateway Mobile Service Switching Centre (G-MSC)*
Merupakan *switch* pada *point* dimana UMTS PLMN dihubungkan ke jaringan CS *eksternal*. Semua CS yang masuk dan keluar harus berhubungan melalui G-MSC.
4. *Serving GPRS (General Packet Radio Service) Support Node (SGSN)*

Mempunyai fungsi sama dengan MSC/VLR tetapi khusus digunakan untuk layanan *packet switch* (PS).

5. *Gateway GPRS Support Node* (GGSN)

Fungsinya mirip dengan G-MSC tetapi untuk layanan *packet switching*.

3.4.2 *UMTS Terrestrial Radio Access Network (UTRAN)*

UTRAN berisi beberapa sub sistem dan elemen jaringan dengan fungsinya masing-masing sebagai berikut:

1. *Radio Network Controller (RNC)*

RNC ini memiliki *level* sama dengan BSC pada GSM, berfungsi untuk mengontrol sejumlah *node B* pada UMTS dan sebagai *interface* ke arah MSC yang berada pada CN-RNC juga bertanggung jawab terhadap proses *handover*

2. *Node B*

Node B seperti halnya BTS pada GSM, bertanggung jawab dalam transmisi radio, perangkat radio dan perangkat *interface* transmisi yang diperlukan di dalam *radio site*. *Node B* dihubungkan ke RNC oleh *interface* Iub.

3.4.3 *User Equipment (UE)*

Bagian ini meliputi berbagai jenis perangkat dengan beragam fungsi. UE mempunyai dua bagian utama yaitu :

1. *Mobile Equipment (ME)*

Merupakan terminal radio yang digunakan untuk komunikasi radio dengan jaringan UMTS melalui Uu. ME masih terbagi menjadi dua yaitu, *Mobile Termination (MT)* dan *Terminal Equipment (TE)*.

2. *UMTS Subscriber Identity Module (USIM)*

Merupakan kartu pintar yang menangani identitas pelanggan, melakukan pengesahan algoritma dan beberapa informasi layanan yang dibutuhkan di terminal.

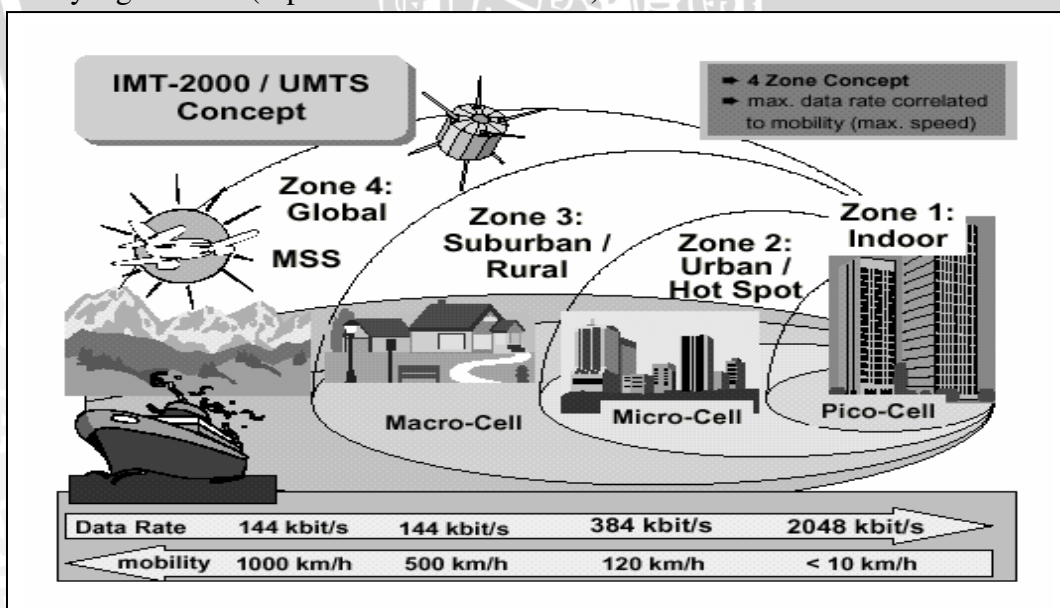
3.5 Daerah Layanan dan Struktur Sel UMTS

UMTS menawarkan cakupan radio global meliputi seluruh dunia. Untuk itu *UMTS Radio Access Network* (URAN) dibangun dengan cara *hierarchical* dalam *layer* bermacam-macam cakupan. Sebuah *layer* tertinggi akan melingkupi suatu area yang geografis lebih besar dibanding sebuah *layer* yang lebih rendah.

Di dalam *layer* paling tinggi akan ada satelit yang mencakup keseluruhan bumi, lapisan yang lebih rendah membentuk *UMTS Terrestrial Radio Access Network* (UTRAN). Mereka dibagi menjadi makro, mikro dan *pico* layer. Masing-masing *layer* dibagi menjadi sel. *Level* hierarki yang terendah adalah sel yang terkecil.

Sel terkecil melayani kepadatan *user* yang tinggi. Oleh karena itu *macro cells* digunakan untuk cakupan wilayah yang luas, penambahan *micro cells* dipasang di dalam area dengan kepadatan penduduk tertinggi dan *pico cells* di dalam bangunan dan untuk itu disebut "*hot spot*" (Thomas. Neubauer, 1998).

Spektrum yang dapat digunakan oleh UMTS adalah terbatas. Penanganan lingkungan yang berbeda-beda dengan berbagai macam kepadatan pengguna *mobile* memerlukan beberapa tipe dan ukuran sel. Untuk tujuan inilah maka *service area* UMTS dibagi kedalam 4 zona dengan cakupan, mobilitas, dan *data rate* yang berbeda (seperti dalam Gambar 3.3).



Gambar 3.3 Zona global UMTS

Sumber : Ericsson

Zone 1: Indoor, Outdoor dengan cakupan yang kecil

Zone 1 sesuai dengan area layanan dalam ruangan dengan terminal yang bersifat tetap atau kecepatan pejalan kaki (kurang dari 10 km/jam) dan pemakai sangat banyak. Pada cakupan kecepatan sangat rendah ini, data transmisi akan mencakup sampai 2048 kbit/s. Area layanan UMTS pada zone 1 disebut *pico cell*, yang area layanannya adalah 10 m di sekitar *base station*.

Contoh area layanan ini adalah kantor, lantai dari suatu bangunan, wilayah hunian pribadi, permintaan yang dituntut adalah hubungan pembicaraan dengan kualitas yang sangat tinggi, *video telephony* dan *video conference*, *transfer* dengan ukuran *file* yang besar, akses *internet* dan *intranet*.

Zone 2: Urban

Zone 2 didasarkan pada penyediaan layanan area urban di daerah *hotspot* dengan pemakai banyak (jumlah pemakai kurang dari zona 1). Cakupan mobilitas tidak berubah, yaitu mencakup kecepatan pejalan kaki (kurang dari 10 km/jam) atau kecepatan rendah 120 km/jam. Pada cakupan kecepatan ini, data transmisi mencakup sampai 384 kbit/s. Area layanan UMTS pada zone 2 disebut *micro cell*, melayani area pada beberapa 100 m di sekitar *base station*. Contoh dari area layanan ini adalah *airport terminal*, stasiun kereta api, bursa saham, tempat publik, dan area kota besar. Layanan yang diminta sama dengan zone 1 ditambah *telematic service* dan informasi lokasi.

Zone 3: Sub Urban / Rural

Zone 3 melingkupi area urban, suburban, dan rural dengan kepadatan pemakai yang tidak terlalu banyak. Cakupan mobilitas adalah kecepatan di atas 120 km/jam atau kecepatan tinggi dengan maksimum 500 km/jam. Transmisi data mencakup sampai 384 kbit/s atau 144 kbit/s. Area layanan UMTS pada zone 3 disebut *macro cell* yang dapat diperbandingkan dengan sel GSM yang normal. Layanan area adalah beberapa km di sekitar *base station*. Zone 3 akan digunakan sebanding dengan teknologi seluler sekarang termasuk layanan untuk semua trafik *terrestrial*.

Zone 4: Global

Zone 4 meliputi seluruh area yang sangat luas yang tidak dicakup oleh zone 1–3, yaitu. area pedesaan, pegunungan, gurun pasir dan bahari pada kepadatan *user* yang rendah. Permintaan mobilitas dalam zone 4 sangat berbeda dan terbentang dari yang tetap (bangunan tunggal, *remote measurement station*) melalui kecepatan medium (*terrestrial*, kendaraan bermotor, kapal laut) sampai kecepatan paling tinggi dengan 1000 km/jam pada maksimum (pesawat udara). Transmisi data mencakup sampai 144 kbit/s. Zone 4 disebut zone Global dan akan dilayani oleh sistem satelit bergerak (MSS).

Adapun tipe-tipe sel yang ada adalah terlihat dalam tabel 3.2 (UMTS Forum).

Tabel 3.2 Tipe Sel dan Karakteristiknya

Tipe Sel	Daerah Operasi	Jari-jari (km)	Mobilitas (kecepatan)	Kec. Max Data
<i>Macro</i>	Urban, Sub urban dan Rural	1-30	Tinggi	384 Kbps
<i>Micro</i>	Urban dan Pusat Kota	0,02-0,4	Tinggi/Rendah	384 Kbps
<i>Pico cell</i>	Ruangan	0,001-0,1	Rendah	2 Mbps

Sumber : UMTS Forum

IMT 2000 (International Mobile Telecommunication 2000) menargetkan laju data paling lambat 144 kbps / 384 kbps untuk cakupan area yang luas dengan mobilitas tinggi, dan 2 Mbps untuk cakupan lokal dengan mobilitas rendah. Berikut ini syarat layanan *Bearer* IMT 2000 seperti yang ditunjukkan pada tabel 3.3.

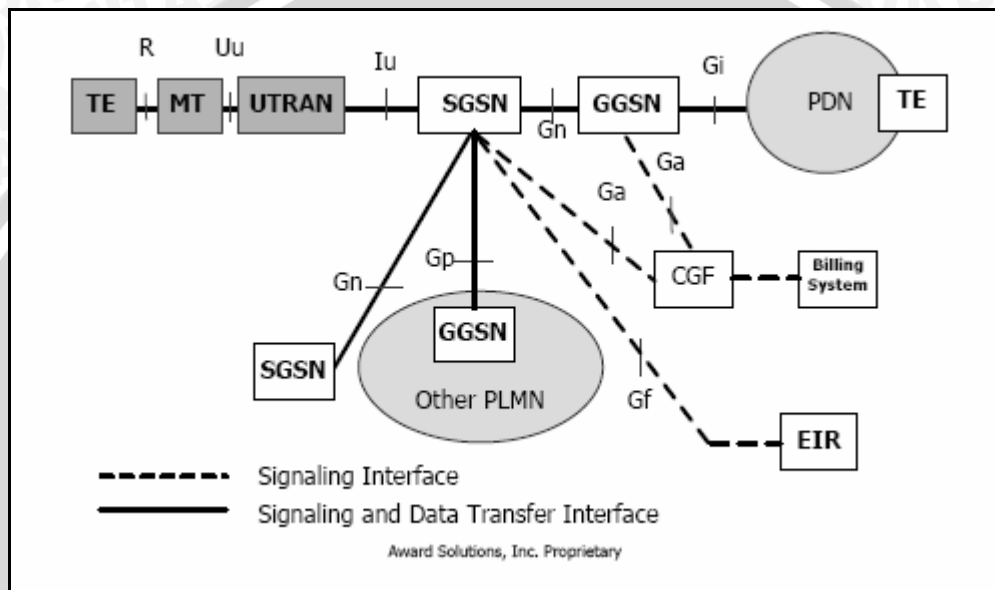
Tabel 3.3 Syarat Layanan *Bearer* IMT 2000

Lingkungan Operasi	<i>Real Time</i>		<i>Non Real Time</i>	
	<i>Peak Bit Rate</i>	<i>BER / Maks Transfer Delay</i>	<i>Peak Bit Rate</i>	<i>BER / Maks Transfer Delay</i>
<i>Rural Outdoor</i> , kec terminal maks 250 km/h	144 kbps	<i>Delay</i> 20 – 300 ms BER $10^{-3} - 10^{-7}$	144 kbps	<i>Delay</i> 150 ms atau lebih BER $10^{-5} - 10^{-8}$
<i>Urban Outdoor</i> , kec terminal maks 150 km/h	384 kbps	<i>Delay</i> 20 – 300 ms BER $10^{-3} - 10^{-7}$	384 kbps	<i>Delay</i> 150 ms atau lebih BER $10^{-5} - 10^{-8}$
<i>In door</i> , kec terminal maks 10 km/h	2 Mbps	<i>Delay</i> 20 – 300 ms BER $10^{-3} - 10^{-7}$	2 Mbps	<i>Delay</i> 150 ms atau lebih BER $10^{-5} - 10^{-8}$

Sumber : Wahyuwijaya, 2004:17

3.6 Konfigurasi Jaringan Paket Data UMTS

Untuk komunikasi bergerak dengan jaringan UMTS, pengimplementasian jaringan paket data dilakukan dengan jalan menghubungkan *mobile equipment* (ME) yang terdiri dari *terminal equipment* (TE) dan *mobile terminal* (MT) dengan UTRAN, SGSN, GGSN dan PDN (*Packet Data Network*) sebagai *server*. Model layanan paket data berbasis IP pada jaringan UMTS ini di tunjukkan pada gambar 3.4 sebagai berikut :



Gambar 3.4 Konfigurasi jaringan paket data UMTS
Sumber : www.awardsolution.com, 2001

3.7 Konfigurasi Aplikasi Videophone pada Jaringan UMTS

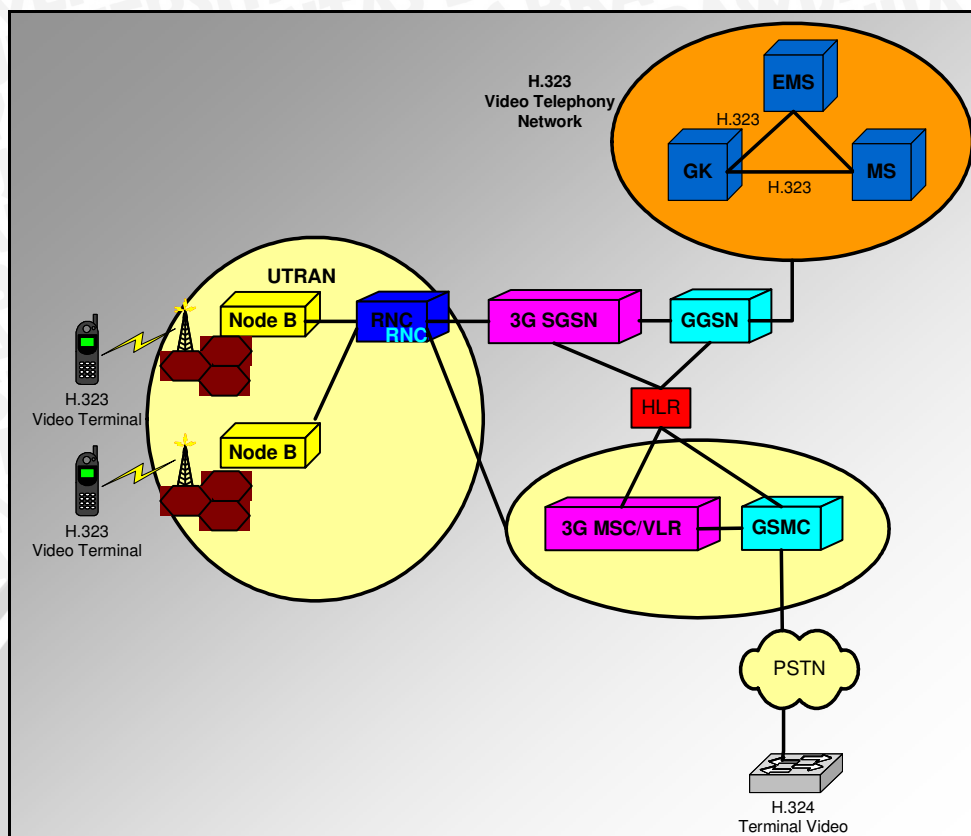
Konfigurasi aplikasi *videophone* pada jaringan UMTS di tunjukkan pada gambar 3.5.

➤ **EMS (Element Management System)**

EMS berfungsi sebagai pembentuk hubungan antara jaringan H.323 dengan GGSN melalui *gatekeeper* dan mengatur hubungan antara GK (*gatekeeper*) dengan *media server*.

➤ **Media server**

Media server sendiri berfungsi sebagai penyedia layanan aplikasi multimedia seperti *videophone*.



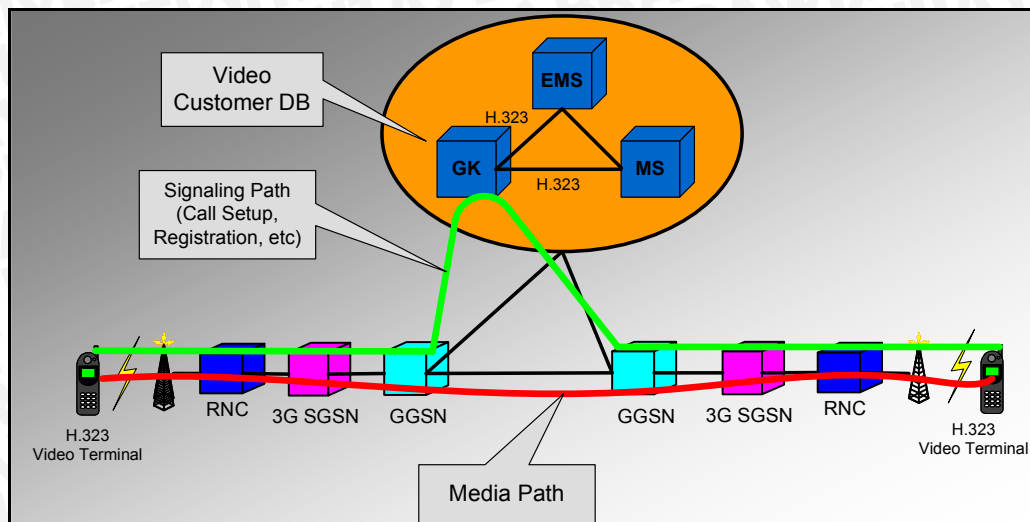
Gambar 3.5 Konfigurasi aplikasi *videophone* pada jaringan UMTS

Sumber : www.INNOACE.com

3.8 Mekanisme Pengiriman Paket Data *Videophone*

Pada gambar 3.6 diperlihatkan alur panggilan *videophone* untuk hubungan antar MS.

1. Jalur *signalling* (*signalling path*) merupakan jalur untuk proses *call setup* dan registrasi yang dilakukan oleh GK (*gatekeeper*) berdasar pada *database* pengguna aplikasi *videophone* yang juga tersimpan pada GK.
2. Setelah GK mengecek data pengguna (*user*), maka GK akan memberikan *signalling* pada GGSN.
3. GGSN akan meneruskan paket *videophone* melalui jalur media (*media path*) menuju GGSN yang dituju dengan *interface* Gi.

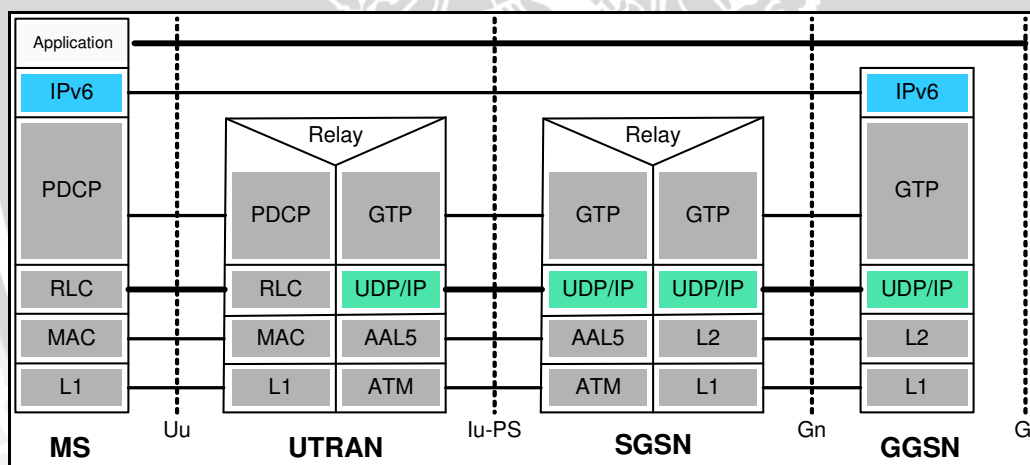


Gambar 3.6 Alur panggilan videophone pada jaringan UMTS

Sumber : Analisis

3.9 Konfigurasi Protocol Stack Jaringan UMTS berbasis IPv6

Protocol Stacks pada jaringan dapat digambarkan pada gambar 3.7



Gambar 3.7 Konfigurasi protokol IPv6 pada jaringan UMTS

sumber : www.nokia.com

Physical layer bertanggung jawab untuk transmisi data pada *air interface*.

Metode akses yang digunakan yaitu FDD dan TDD W-CDMA (UMTS Re.'99). Alasan utama UMTS menggunakan dua metode akses jamak adalah untuk meningkatkan kualitas layanan ketika terjadi kepadatan trafik yang tinggi di area pikosel dan mikrosel serta pengaksesan data asimetris (tidak sama) antara arah



transmisi *uplink* dan *downlink*, seperti layanan *internet*, *audio-video*, *multimedia*, dan *videoconference*.

Medium Access Control (MAC) memiliki tugas-tugas antara lain *transfer data*, menyeleksi secara tepat kecepatan bit dan kode setiap unit informasi yang dikirimkan menuju *physical layer*, *traffic monitoring*, dan *multiplexing*

Radio Link Control (RLC) berfungsi menghubungkan layer 2 menuju layer di atasnya, menyalin data, *transfer data*, dan deteksi kesalahan transmisi yang gagal dari *physical layer*.

Paket Data Convergence Protocol (PDCP) berfungsi berfungsi untuk transmisi dan menerima *protocol data units (PDUs)* pada *layer network*, mengirimkan beberapa protokol yang berbeda pada *network layer*, meningkatkan efisiensi kanal.

GPRS Tunnelling Protocol (GTP) berfungsi untuk membuat jalur atau aliran data antara UTRAN dan SGSN, dan antara *backbone* GSNs.

Asynchronous Transfer Mode (ATM), informasi dikirim dalam *fixed-size cells (53 byte)*.

ATM Adaptation Layer 5 (AAL5), protokol ini menyediakan dukungan untuk *variable-bitrate connection-oriented* atau *connectionless data services*.

3.10 Parameter-parameter yang dianalisis

3.10.1 Delay End-to-End

Salah satu ukuran unjuk kerja jaringan data adalah *delay* yang dibutuhkan untuk mengirimkan data dari sumber ke tujuannya. Pertimbangan *delay* tersebut mempengaruhi penentuan dan unjuk kerja dari algoritma jaringan, seperti perutean dan *flow control*. Definisi umum dari *delay* adalah waktu tunda. Dalam jaringan berbasis *packet switching*, *delay* yang terjadi merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari sumber ke tujuannya pada setiap *hop*. Sesuai dengan persamaan 2.13 bahwa *delay end-to-end* untuk aplikasi *videophone* adalah :

$$t_{\text{tot}} = t_{\text{CODEC}} + t_{\text{gk}} + t_{\text{net}}$$

dimana t_{net} (*delay* jaringan) didapat dengan persamaan:

$$t_{\text{net}} = t_{\text{proc}} + t_{\text{w}} + t_{\text{T}} + t_{\text{p}} \quad (3.1)$$

dimana:

t_{net}	=	<i>delay</i> jaringan (detik)
t_{proc}	=	<i>delay</i> proses (detik)
t_w	=	<i>delay</i> antrian (detik)
t_T	=	<i>delay</i> transmisi (detik)
t_p	=	<i>delay</i> propagasi (detik)

3.10.1.1 Delay Proses

Delay proses adalah waktu yang dibutuhkan untuk memproses paket data dan untuk menentukan kemana data tersebut akan diteruskan. *Delay* proses pada jaringan UMTS berupa *delay* enkapsulasi dan *delay* dekapsulasi. Yang dimaksud *delay* enkapsulasi/ dekapsulasi adalah banyaknya waktu yang dibutuhkan untuk mengubah *segment* TCP/UDP menjadi paket data untuk dilewatkan pada jaringan.

Pada bagian ini dilakukan analisa proses enkapsulasi/dekapsulasi dari masing-masing protokol tiap *layer* jaringan seperti pada gambar 3.7, yaitu dengan cara menambahkan *header* dan perubahan format data mulai dari MT (*Mobile Terminal*) / MS (*Mobile Station*) hingga mencapai tujuan GGSN (*Gateway GPRS Support Node*). Namun, apabila paket data yang akan dienkapsulasi panjangnya melebihi kapasitas transfer maksimum dari suatu protokol, maka paket data tersebut sebelumnya akan disegmentasi/dipecah sesuai dengan ukuran kapasitas transfer maksimum dari protokol tersebut. Setelah sampai di penerima, maka paket data yang terenkapsulasi tersebut akan didekapsulasi untuk mendapatkan kembali data aslinya.

Adapun *delay* enkapsulasi/dekapsulasi untuk masing-masing *node* sesuai dengan model jaringan adalah :

➤ *Mobile Terminal forward*

Dalam perhitungan adalah menganggap bahwa besarnya data adalah setelah melewati *layer* aplikasi dan menuju ke *transport layer*. Sehingga data akan menjadi paket yang disebut *segment*. Besarnya *segment* adalah ditentukan sesuai dengan persamaan berikut:

$$W_{segmen} = W_{data} + \text{Header}_{UDP/TCP} \quad (3.2)$$

dengan :

W_{segmen} = panjang segmen TCP/UDP (byte)

W_{data} = jumlah *application data*/data sebelum terenkapsulasi (byte)

Header = panjang *header* TCP (20 byte) atau UDP (8 byte)

Pada *layer transport*, suatu segmen dapat ditambahkan header UDP atau TCP. Menggunakan UDP jika *protocol layer transport* tersebut digunakan untuk membawa aplikasi yang bersifat *real time* dan TCP jika menggunakan aplikasi *non real time*. Pada skripsi ini karena aplikasi yang digunakan adalah *videophone* maka yang digunakan adalah UDP yang bersifat *real time*. Dari *layer transport* kemudian data dibawa menuju *network layer*, yaitu IP untuk diubah menjadi *datagram* dengan penambahan *header* IP. Karena pada skripsi ini IP yang digunakan adalah IPv6 maka ditambahkan *header* IPv6 sebesar 40 byte sesuai persamaan berikut:

$$W_{\text{datagram}} = W_{\text{segmen}} + \text{Header}_{\text{IPv6}} + \text{Header ICMP} \quad (3.3)$$

dimana :

W_{datagram} = panjang *datagram* IP (byte)

$\text{Header}_{\text{IPv6}}$ = panjang *header* IPv6 (40 byte)

Header ICMP = panjang *header* ICMP (8 byte)

Setelah didapatkan nilai *datagram* IP kemudian data menuju *datalink layer*. *Data link layer* pada MT menggunakan PDCP (*Packet Data Convergence Protocol*). Pada *link layer* PDCP *datagram* diubah menjadi PDCP Data PDU (*Protocol Data Unit*) sesuai dengan persamaan 3.4 dan format seperti pada gambar 3.8 yaitu dengan menambahkan *header* PDU Type sebesar 3 bits dan PID (*Packet Identifier*) sebesar 5 bits jadi total *header* sebesar 8 bits atau 1 byte (3GPP TS 25.323 Rel.5).

PDU Type	PID
Data	

Gambar 3.8 Format PDCP data PDU
Sumber : 3GPP TS 25.323 Rel.5

$$W_{\text{PDCP}} = W_{\text{datagram}} + \text{Header}_{\text{PDCP}} \quad (3.4)$$

dengan :

W_{PDCP} = panjang data PDCP (byte)

$\text{Header}_{\text{PDCP}}$ = *header* PDCP (1 byte)

Jika jumlah *datagram* IP ini melebihi jumlah MTU (*Maximum Transmission Unit*) sebesar 1500 byte maka *datagram* mengalami segmentasi menjadi beberapa buah *frame* sesuai persamaan berikut:

$$N_{frame-PDCP} = \frac{W_{datagram}}{MTU_{PDCP}} \quad (3.5)$$

dimana :

$N_{frame-PDCP}$ = Jumlah *Frame PDCP*

MTU_{PDCP} = *Maximum Transmission Unit PDCP* (1500 byte)

Kemudian dengan menggunakan persamaan dibawah ini diperoleh panjang *frame PDCP* sebesar:

$$W_{frame PDCP} = MTU_{PDCP} + Header_{PDCP} \quad (3.6)$$

dengan :

$W_{frame PDCP}$ = panjang *frame PDCP* (byte)

$Header_{PDCP}$ = *header PDCP* (1 byte)

Sehingga jumlah total *frame PDCP* yang dapat dikirimkan ke RLC (*Radio Link Control*) yaitu:

$$W_{frame PDCP total} = N_{frame PDCP} \times W_{frame PDCP} \quad (3.7)$$

dengan :

$W_{frame PDCP total}$ = panjang *total frame* pada PDCP

Pada *layer RLC*, PDCP Data PDU akan disegmentasi menjadi RLC PDU *fixed size*. Pada dokumen 3GPP TS 25.322 Rel.5 tentang *Radio Link Control (RLC) Protocol Specification* tidak ditegaskan besaran untuk tiap *fixed size* tetapi pada skripsi ini besaran tiap *fixed size* mengacu pada besaran yang digunakan pada penelitian di *Ericsson EuroLab Netherlands* yaitu sebesar 40 byte. Sehingga jumlah *frame RLC PDU* yaitu :

$$N_{frame-RLC} = \frac{W_{PDCP}}{40byte} \quad (3.8)$$

dimana :

$N_{frame-RLC}$ = Jumlah *frame RLC PDU*

W_{PDCP} = panjang data PDCP (byte)

Kemudian dengan persamaan 3.9 diperoleh panjang *frame RLC* sebesar :

$$W_{frame\ RLC} = 40\ \text{byte} + \text{Header RLC} \quad (3.9)$$

dengan :

$$W_{frame\ RLC} = \text{panjang frame RLC PDU (byte)}$$

$$\text{Header RLC} = \text{header RLC (2 byte)}$$

Sehingga jumlah total *frame* RLC PDU yang dikirimkan ke MAC adalah :

$$W_{total\ frame\ RLC} = N_{frame\ RLC} \times W_{frame\ RLC} \quad (3.10)$$

dengan :

$$W_{total\ frame\ RLC} = \text{jumlah total frame RLC PDU (byte)}$$

Pada *layer* MAC, untuk menjadi packet data yang siap untuk ditransmisikan lewat *physical layer* maka panjang *frame* RLC PDU akan ditambah *header* MAC sebesar 3 byte dan CRC 2 byte.

$$W_{frame\ MAC} = \text{Header MAC} + W_{frame\ RLC} + \text{CRC} \quad (3.11)$$

dengan :

$$W_{frame\ MAC} = \text{panjang frame MAC PDU (byte)}$$

$$\text{Header MAC} = \text{header MAC (3 byte)}$$

$$\text{CRC} = \text{Cyclic Redundancy Check (2 byte)}$$

Sehingga jumlah total *frame* MAC PDU yang siap untuk ditransmisikan lewat *physical layer* adalah :

$$W_{total\ frame\ MAC} = N_{frame\ RLC} \times W_{frame\ MAC} \quad (3.12)$$

dengan :

$$W_{total\ frame\ MAC} = \text{jumlah total frame MAC PDU (byte)}$$

Nilai *delay* enkapsulasi pada MT saat lintasan *forward* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E1} = \frac{W_{total\ frame\ MAC} \times 8}{V_{MT}} \quad (3.13)$$

dimana :

$$t_{E1} = \text{delay enkapsulasi di MT saat lintasan forward (dtk)}$$

$$V_{MT} = \text{kecepatan transmisi data di MT (bps)}$$

➤ **UTRAN (UMTS Terrestrial Radio Access Network) forward**

Pada UTRAN menggunakan *interface* STM-1 dengan kecepatan transmisi adalah 155 Mbps. Nilai *delay* dekapsulasi pada UTRAN saat lintasan *forward* dapat dihitung dengan persamaan :



$$t_{D1} = \frac{W_{totalframeMAC}}{V_{UTRAN}} \quad (3.14)$$

dengan :

t_{D1} = delay dekapsulasi di UTRAN saat lintasan *forward* (detik)

V_{UTRAN} = kecepatan transmisi data di UTRAN (bps)

Pada UTRAN *frame* MAC PDU disegmentasi menjadi *frame* GTP sesuai dengan MTU yaitu 1500 byte. Kemudian untuk mendapatkan jumlah *frame* pada GTP adalah dengan persamaan :

$$N_{frameGTP} = \frac{W_{totalframeMAC}}{MTU} \quad (3.15)$$

dengan :

$N_{frameGTP}$ = Jumlah *Frame GTP*

MTU_{GTP} = *Maximum Transmission Unit GTP* (1500 byte)

Selanjutnya masing-masing *frame* tersebut dienkapsulasi dengan menambahkan *header* GTP, UDP dan IP (3GPP TS 29.060 Rel.5) sebelum ditransmisikan ke *layer* AAL5 yaitu sesuai dengan persamaan :

$$W_{frameGTP} = MTU_{GTP} + headerGTP + headerUDP + headerIP \quad (3.16)$$

dengan :

$W_{frameGTP}$ = panjang *frame* GTP (byte)

header GTP = *header* GTP (8 byte)

header UDP = *header* UDP (8 byte)

header IP = *header* IPv6 (40 byte)

Sehingga jumlah *total frame* GTP yang siap untuk ditransmisikan lewat AAL5 adalah :

$$W_{totalframeGTP} = N_{frameGTP} \times W_{frameGTP} \quad (3.17)$$

Apabila *frame* MAC PDU yang diterima tidak melebihi MTU 1500 byte maka *frame* tersebut langsung dienkapsulasi dengan menambahkan *header* GTP, UDP dan IP yaitu sesuai dengan persamaan 3.18 :

$$W_{GTP} = W_{totalframeMAC} + headerGTP + headerUDP + headerIP \quad (3.18)$$

Pada *layer* AAL5 *frame* GTP akan diberi *header* sebelum disegmentasi menjadi *fixed-size cells* (53 byte) pada *layer* ATM. Jumlah *header* yang akan diberikan pada *frame* GTP terdiri dari *header* PAD dengan ukuran maksimum

sebesar 48 byte dan *header* CT (CPCS(*Common Part Convergence Sublayer*) *Trailer*) sebesar 8 byte. Sehingga besar paket AAL5 yang siap untuk ditransmisikan pada *layer* ATM adalah :

$$W_{AAL5} = W_{total\ frame\ GTP} + \text{header PAD} + \text{header CT} \quad (3.19)$$

dengan :

$$W_{AAL5} = \text{besar paket AAL5 (byte)}$$

$$\text{header PAD} = \text{header PAD (48 byte)}$$

$$\text{header CT} = \text{header CT (8 byte)}$$

Pada *layer* ATM paket AAL5 akan disegmentasi menjadi *fixed-size cells* ATM sebesar 53 byte yang terdiri dari *payload* ATM sebesar 48 byte dan *header* ATM sebesar 8 byte (Stallings). Kemudian untuk mendapatkan jumlah *frame payload* ATM adalah dengan persamaan :

$$N_{frame\ ATM} = \frac{W_{AAL5}}{48\text{byte}} \quad (3.20)$$

dengan :

$$N_{frame\ ATM} = \text{jumlah } frame\ \text{payload ATM}$$

$$W_{AAL5} = \text{besar paket AAL5 (byte)}$$

Sehingga jumlah *total frame* ATM yang siap untuk ditransmisikan ke SGSN adalah :

$$W_{total\ frame\ ATM} = N_{frame\ ATM} \times \text{fixed-sizes cells ATM} \quad (3.21)$$

dengan :

$$W_{total\ frame\ ATM} = \text{jumlah } total\ frame\ ATM$$

$$\text{fixed-sizes cells ATM} = 53\ \text{byte}$$

Nilai *delay* enkapsulasi pada UTRAN saat lintasan *forward* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E2} = \frac{W_{total\ frame\ ATM} \times 8}{V_{UTRAN}} \quad (3.22)$$

dimana :

$$t_{E2} = \text{delay enkapsulasi di UTRAN saat lintasan } forward\ (dtk)$$

$$V_{UTRAN} = \text{kecepatan transmisi data di UTRAN (bps)}$$

➤ **SGSN (*Serving GPRS Support Node*) forward**

Pada SGSN paket yang diterima dari UTRAN akan mengalami proses dekapsulasi. Pada skripsi ini menggunakan SGSN dengan teknologi *Gigabit Ethernet* dengan transmisi data sebesar 1,25 Gbps. Nilai *delay* dekapsulasi pada SGSN saat lintasan *forward* didapatkan dengan persamaan :

$$t_{D2} = \frac{W_{totalframeATM}}{V_{SGSN}} \quad (3.23)$$

dengan :

t_{D2} = *delay* dekapsulasi di SGSN saat lintasan *forward* (detik)

V_{SGSN} = kecepatan transmisi data di SGSN (bps)

Frame yang akan dikirimkan dari SGSN ke GGSN akan mengalami proses enkapsulasi dengan menambahkan *header* GTP, UDP, IP (3GPP TS 29.060 Rel.5) dan CRC pada L2 sebelum ditransmisikan ke GGSN dengan persamaan 3.24 :

$$W_{SGSN} = W_{total\ frame\ ATM} + \text{header GTP} + \text{header UDP} + \text{header IP} + \text{CRC} \quad (3.24)$$

dengan :

W_{SGSN} = panjang paket SGSN (byte)

header GTP = *header* GTP (8 byte)

header UDP = *header* UDP (8 byte)

header IP = *header* IPv6 (40 byte)

CRC = *Cyclic Redundancy Check* (2 byte)

Apabila *frame* yang dikirim dari UTRAN melebihi MTU *Ethernet* maka pada SGSN akan dilakukan proses segmentasi sesuai dengan MTU *Ethernet* yaitu 1500 byte. Kemudian untuk mendapatkan jumlah *frame* pada SGSN adalah dengan persamaan :

$$N_{frameSGSN} = \frac{W_{totalframeATM}}{MTU_{Ethernet}} \quad (3.25)$$

dengan :

$N_{frame\ SGSN}$ = Jumlah *Frame* SGSN

$MTU_{Ethernet}$ = *Maximum Transmission Unit* *Ethernet* (1500 byte)

Selanjutnya masing-masing *frame* tersebut dienkapsulasi dengan menambahkan *header* GTP, UDP, IP dan CRC pada L2 sebelum ditransmisikan ke SGSN yaitu sesuai dengan persamaan :

$$W_{frameSGSN} = MTU_{Ethernet} + header\ GTP + header\ UDP + header\ IP + CRC \quad (3.26)$$

Sehingga jumlah *total frame* SGSN yang siap untuk ditransmisikan ke SGSN adalah :

$$W_{total\ frame\ SGSN} = N_{frame\ SGSN} \times W_{frameSGSN} \quad (3.27)$$

dengan :

$$W_{total\ frame\ SGSN} = \text{jumlah } total\ frame\ SGSN$$

$$W_{frameSGSN} = \text{panjang } frame\ SGSN$$

Nilai *delay* enkapsulasi pada SGSN saat lintasan *forward* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E3} = \frac{W_{totalframeSGSN} \times 8}{V_{SGSN}} \quad (3.28)$$

dimana :

$$t_{E3} = \text{delay enkapsulasi di SGSN saat lintasan } forward \text{ (dtk)}$$

$$V_{SGSN} = \text{kecepatan transmisi data di SGSN (bps)}$$

➤ GGSN (*Gateway GPRS Support Node*)

Pada GGSN paket yang diterima dari SGSN akan mengalami proses dekapsulasi. Pada skripsi ini GGSN yang digunakan yaitu dari Cisco dengan type *Cisco 7600 Series Router* dengan teknologi *Gigabit Ethernet* dengan transmisi data sebesar 10 Gbps. Nilai *delay* dekapsulasi pada GGSN saat lintasan *forward* didapatkan dengan persamaan :

$$t_{D3} = \frac{W_{totalframeSGSN}}{V_{GGSN}} \quad (3.29)$$

dengan :

$$t_{D3} = \text{delay dekapsulasi di GGSN saat lintasan } forward \text{ (detik)}$$

$$V_{GGSN} = \text{kecepatan transmisi data di GGSN (bps)}$$

Setelah *frame* melalui semua *node* pada jaringan UMTS untuk lintasan *forward*, selanjutnya dilakukan analisis dan perhitungan untuk lintasan *reverse*

yaitu dari GGSN (*Gateway GPRS Support Node*) menuju MT (*Mobile Terminal*) penerima.

Frame yang akan dikirimkan dari GGSN ke SGSN akan mengalami proses enkapsulasi dengan menambahkan *header* GTP, UDP, IP (3GPP TS 29.060 Rel.5) dan CRC pada L2 sebelum ditransmisikan ke SGSN sehingga diperoleh :

$$W_{GGSN} = W_{total\ frame\ SGSN} + header\ GTP + header\ UDP + header\ IP + CRC \quad (3.30)$$

dengan :

- W_{GGSN} = panjang paket GGSN (byte)
- header* GTP = *header* GTP (8 byte)
- header* UDP = *header* UDP (8 byte)
- header* IP = *header* IPv6 (40 byte)
- CRC = *Cyclic Redundancy Check* (2 byte)

Apabila *frame* yang dikirim dari SGSN melebihi MTU *Ethernet* maka pada GGSN akan dilakukan proses segmentasi sesuai dengan MTU *Ethernet* yaitu 1500 byte. Kemudian untuk mendapatkan jumlah *frame* pada GGSN adalah dengan persamaan :

$$N_{frame\ GGSN} = \frac{W_{total\ frame\ SGSN}}{MTU_{Ethernet}} \quad (3.31)$$

dengan :

- $N_{frame\ GGSN}$ = Jumlah *Frame* GGSN
- $MTU_{Ethernet}$ = *Maximum Transmission Unit Ethernet* (1500 byte)

Selanjutnya masing-masing *frame* tersebut dienkapsulasi dengan menambahkan *header* GTP, UDP, IP dan CRC pada L2 sebelum ditransmisikan ke SGSN yaitu sesuai dengan persamaan :

$$W_{frame\ GGSN} = MTU_{Ethernet} + header\ GTP + header\ UDP + header\ IP + CRC \quad (3.32)$$

Sehingga jumlah *total frame* GGSN yang siap untuk ditransmisikan ke SGSN adalah :

$$W_{total\ frame\ GGSN} = N_{frame\ GGSN} \times W_{frame\ GGSN} \quad (3.33)$$

dengan :

$$W_{total\ frame\ GGSN} = \text{jumlah total frame GGSN}$$

$$W_{frame\ GGSN} = \text{panjang frame GGSN}$$

Sehingga nilai *delay* enkapsulasi pada GGSN saat lintasan *reverse* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E4} = \frac{W_{total\ frame\ GGSN} \times 8}{V_{GGSN}} \tag{3.34}$$

dimana :

$$t_{E4} = \text{delay enkapsulasi di GGSN saat lintasan reverse (dtk)}$$

$$V_{GGSN} = \text{kecepatan transmisi data di GGSN (bps)}$$

➤ **SGSN (*Serving GPRS Support Node*) reverse**

Pada SGSN paket yang diterima dari GGSN akan mengalami proses dekapsulasi. Nilai *delay* dekapsulasi saat lintasan *reverse* pada SGSN didapatkan dengan persamaan :

$$t_{D4} = \frac{W_{total\ frame\ GGSN}}{V_{SGSN}} \tag{3.35}$$

dengan :

$$t_{D4} = \text{delay dekapsulasi di SGSN saat lintasan reverse (detik)}$$

$$V_{SGSN} = \text{kecepatan transmisi data di SGSN (bps)}$$

Pada SGSN *frame* GGSN disegmentasi menjadi *frame* GTP sesuai dengan MTU yaitu 1500 byte. Kemudian untuk mendapatkan jumlah *frame* pada GTP adalah dengan persamaan :

$$N_{frame\ GTP} = \frac{W_{total\ frame\ GGSN}}{MTU} \tag{3.36}$$

dengan :

$$N_{frame\ GTP} = \text{Jumlah Frame GTP}$$

$$MTU_{GTP} = \text{Maximum Transmission Unit GTP (1500 byte)}$$

Selanjutnya masing-masing *frame* tersebut dienkapsulasi dengan proses yang sama seperti pada persamaan 3.16 – 3.17.

Apabila *frame* GGSN yang diterima tidak melebihi MTU 1500 byte maka *frame* tersebut langsung dienkapsulasi dengan menambahkan *header* GTP, UDP dan IP yaitu sesuai dengan persamaan 3.37 :

$$W_{GTP} = W_{total\ frame\ GGSN} + \text{header GTP} + \text{header UDP} + \text{header IP} \tag{3.37}$$



Selanjutnya proses enkapsulasi sama seperti pada persamaan 3.19 – 3.21. Nilai *delay* enkapsulasi pada SGSN saat lintasan *reverse* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E5} = \frac{W_{totalframeATM} \times 8}{V_{SGSN}} \quad (3.38)$$

dimana :

t_{E5} = *delay* enkapsulasi di SGSN saat lintasan *reverse* (dtk)

V_{SGSN} = kecepatan transmisi data di SGSN (bps)

➤ **UTRAN (UMTS Terrestrial Radio Access Network) reverse**

Pada UTRAN paket yang diterima dari SGSN akan mengalami proses dekapsulasi. Nilai *delay* dekapsulasi pada UTRAN saat lintasan *reverse* didapatkan dengan persamaan

$$t_{D5} = \frac{W_{totalframeATM}}{V_{UTRAN}} \quad (3.39)$$

dengan :

t_{D5} = *delay* dekapsulasi di UTRAN saat lintasan *reverse* (detik)

V_{UTRAN} = kecepatan transmisi data di UTRAN (bps)

Pada UTRAN *frame* ATM disegmentasi menjadi PDCP Data PDU (*Protocol Data Unit*) sesuai dengan persamaan 3.40 dan format seperti pada gambar 3.8 yaitu dengan menambahkan *header* PDU Type sebesar 3 bits dan PID (*Packet Identifier*) sebesar 5 bits jadi total *header* sebesar 8 bits atau 1 byte (3GPP TS 25.323 Rel.5).

$$W_{PDCP} = W_{total frame ATM} + Header_{PDCP} \quad (3.40)$$

dengan :

W_{PDCP} = panjang data PDCP (byte)

$Header_{PDCP}$ = *header* PDCP (1 byte)

Jika jumlah paket ATM ini melebihi jumlah MTU (*Maximum Transmit Unit*) sebesar 1500 byte maka paket mengalami segmentasi menjadi beberapa buah *frame* sesuai persamaan berikut:

$$N_{frame-PDCP} = \frac{W_{totalframeATM}}{MTU_{PDCP}} \quad (3.41)$$

dimana :

$$N_{\text{frame-PDCP}} = \text{Jumlah Frame PDCP}$$

$$MTU_{\text{PDCP}} = \text{Maximum Transmission Unit PDCP (1500 byte)}$$

Kemudian dengan menggunakan persamaan 3.6 diperoleh panjang *frame* PDCP sebesar:

$$W_{\text{frame PDCP}} = MTU_{\text{PDCP}} + \text{Header}_{\text{PDCP}}$$

Sehingga jumlah total *frame* PDCP yang dapat dikirimkan ke RLC (*Radio Link Control*) sesuai dengan persamaan 3.7 yaitu:

$$W_{\text{frame PDCP total}} = N_{\text{frame PDCP}} \times W_{\text{frame PDCP}}$$

Selanjutnya masing-masing *frame* tersebut dienkapsulasi dengan proses yang sama seperti pada persamaan 3.8 – 3.12. Nilai *delay* enkapsulasi pada UTRAN saat lintasan *reverse* yang terjadi didapatkan dengan persamaan dibawah ini:

$$t_{E6} = \frac{W_{\text{totalframeMAC}} \times 8}{V_{\text{UTRAN}}} \quad (3.42)$$

dimana :

$$t_{E6} = \text{delay enkapsulasi di UTRAN saat lintasan reverse (dtk)}$$

$$V_{\text{UTRAN}} = \text{kecepatan transmisi data di UTRAN (bps)}$$

➤ **Mobile Terminal reverse**

Nilai *delay* dekapsulasi pada MT (*Mobile Terminal*) saat lintasan *reverse* dapat dihitung dengan persamaan :

$$t_{D6} = \frac{W_{\text{totalframeMAC}}}{V_{\text{MT}}} \quad (3.43)$$

dengan :

$$t_{D6} = \text{delay dekapsulasi di MT saat lintasan reverse (detik)}$$

$$V_{\text{MT}} = \text{kecepatan transmisi data di MT (bps)}$$

➤ **Delay enkapsulasi dan dekapsulasi dari MT pengirim ke MT penerima adalah :**

$$t_{E \text{ total}} = t_{E1} + t_{E2} + t_{E3} + t_{E4} + t_{E5} + t_{E6} \quad (3.44)$$

$$t_{D \text{ total}} = t_{D1} + t_{D2} + t_{D3} + t_{D4} + t_{D5} + t_{D6} \quad (3.45)$$

$$t_{\text{proc}} = t_{E \text{ total}} + t_{D \text{ total}} \quad (3.46)$$

3.10.1.2 Delay transmisi

Delay transmisi adalah waktu yang dibutuhkan untuk mentransmisikan bit pertama dari paket data hingga bit terakhir melalui media transmisi tertentu. Dalam analisis ini dilakukan penghitungan *delay* transmisi *end-to-end* yaitu hubungan antara MT-Node B (*Uu/WCDMA Air interface*), Node B-RNC (*Iub interface*), RNC-SGSN (*Iu interface*) dan SGSN-GGSN (*Gn Interface*) yang dihitung dalam dua lintasan yaitu *forward* dan *reverse*. Berikut adalah persamaan untuk masing-masing *delay* transmisi untuk tiap *node* dan tiap lintasan.

(a) *Uu Interface* saat lintasan *forward*

Antara MT dengan Node B dihubungkan dengan menggunakan *Uu interface* dengan kecepatan sesuai dengan *data rate*. Sehingga *delay* transmisi yang terjadi pada *Uu interface* adalah:

$$t_{T1} = \frac{W_{totalframeMAC}}{n \times V_{MT}} \quad (3.47)$$

dengan:

$$\begin{aligned} t_{T1} &= \text{delay transmisi pada } Uu \text{ interface lintasan forward (detik)} \\ n &= \text{jumlah slot tiap MS} \\ V_{MT} &= \text{kecepatan data rate (bps)} \end{aligned}$$

(b) *Iub Interface* saat lintasan *forward*

Iub interface menghubungkan antara Node B dan RNC. *Physical layer* untuk *Iub interface* pada jaringan UMTS berbasis IP menggunakan *interface* sistem transmisi digital STM-1 yang memberikan kecepatan data sebesar 155 Mbps [3GPP TS 25.411].

Sehingga nilai *delay* transmisinya didapatkan dengan persamaan:

$$t_{T2} = \frac{W_{total frame ATM}}{V_{Iub interface}} \quad (3.48)$$

dengan:

$$\begin{aligned} t_{T2} &= \text{delay transmisi pada } Iub \text{ interface lintasan forward (detik)} \\ V_{Iub interface} &= \text{kecepatan transmisi pada } Iub \text{ Interface (bps)} \end{aligned}$$

(c) Iu Interface saat lintasan forward

Antara RNC dengan SGSN dihubungkan dengan menggunakan *interface Gigabit Ethernet* yang memberikan kecepatan data sebesar 1 Gbps. Sehingga *delay* transmisi yang terjadi pada *Iu interface* adalah ditetapkan dengan persamaan:

$$t_{T3} = \frac{W_{totalframeATM}}{V_{Iu\ interface}} \tag{3.49}$$

dengan:

- t_{T3} = *delay* transmisi pada *Iu Interface* saat lintasan *forward* (detik)
- $V_{Iu\ interface}$ = kecepatan transmisi pada *Iu interface* (bps)

(d) Gn Interface saat lintasan forward

Antara SGSN dengan GGSN dihubungkan dengan menggunakan *ethernet* sebagai *interface* yang memiliki kecepatan transmisi data sebesar 1 Gbps. Sehingga *delay* transmisi yang terjadi pada *Gn interface* adalah sesuai dengan persamaan:

$$t_{T4} = \frac{W_{totalframeSGSN}}{V_{ethernet}} \tag{3.50}$$

dengan:

- t_{T4} = *delay* transmisi saat lintasan *forward* pada *Gn interface* (detik)
- $V_{ethernet}$ = kecepatan transmisi pada *Gn interface* (bps)

(e) Gn Interface saat lintasan reverse

Antara GGSN dengan SGSN dihubungkan dengan menggunakan *ethernet* sebagai *interface* yang memiliki kecepatan transmisi data sebesar 1 Gbps. Sehingga *delay* transmisi yang terjadi pada *Gn interface* pada saat lintasan *reverse* ditentukan dengan persamaan:

$$t_{T5} = \frac{W_{totalframeGGSN}}{V_{ethernet}} \tag{3.51}$$

dengan:

- t_{T5} = *delay* transmisi pada *Gn interface* saat lintasan *reverse* (detik)
- $V_{ethernet}$ = kecepatan transmisi pada *Gn interface* (bps)



(f) Iu Interface saat lintasan reverse

Antara SGSN dengan RNC dihubungkan dengan menggunakan *interface* sistem *Gigabit ethernet* yang memberikan kecepatan data sebesar 1Gbps. Sehingga *delay* transmisi yang terjadi pada *Iu interface* adalah didapatkan dengan persamaan:

$$t_{T6} = \frac{W_{totalframeATM}}{V_{Iu\ interface}} \tag{3.52}$$

dengan:

- t_{T6} = *delay* transmisi pada *Iu interface* saat lintasan reverse (detik)
- $V_{Iu\ interface}$ = kecepatan transmisi pada *Iu interface* (bps)

(g) Iub Interface saat lintasan reverse

Iub interface menghubungkan antara RNC dengan Node B. *Physical layer* untuk *Iub interface* pada jaringan UMTS berbasis IP menggunakan *interface* sistem transmisi digital STM-1 yang memberikan kecepatan data sebesar 155 Mbps [3GPP TS 25.411].

Sehingga nilai *delay* transmisinya didapatkan dengan persamaan:

$$t_{T7} = \frac{W_{total\ frame\ ATM}}{V_{Iub\ interface}} \tag{3.53}$$

dengan:

- t_{T7} = *delay* transmisi pada *Iub interface* lintasan reverse (detik)
- $V_{Iub\ interface}$ = kecepatan transmisi pada *Iub Interface* (bps)

(h) Uu Interface saat lintasan reverse

Antara Node B dengan MT dihubungkan dengan menggunakan *Uu interface* dengan kecepatan sesuai *data rate*. Sehingga *delay* transmisi yang terjadi pada *Uu interface* adalah:

$$t_{T8} = \frac{W_{totalframeMAC}}{n \times V_{MT}} \tag{3.54}$$

dengan:

- t_{T8} = *delay* transmisi pada *Uu interface* lintasan reverse (detik)
- n = jumlah slot pada lintasan *forward*

(i) Delay transmisi total

Delay transmisi untuk lintasan *forward* ditentukan dengan persamaan :

$$t_{T \text{ forward}} = t_{T1} + t_{T2} + t_{T3} + t_{T4} \tag{3.55}$$

Delay transmisi untuk lintasan *reverse* ditentukan dengan persamaan :

$$t_{T \text{ reverse}} = t_{T5} + t_{T6} + t_{T7} + t_{T8} \tag{3.56}$$

Delay transmisi total ditentukan dengan persamaan :

$$t_{T \text{ Total}} = t_{T \text{ forward}} + t_{T \text{ reverse}} \tag{3.57}$$

dengan:

$$t_{T \text{ forward}} = \text{delay transmisi untuk lintasan } \textit{forward} \text{ (detik)}$$

$$t_{T \text{ reverse}} = \text{delay transmisi untuk lintasan } \textit{reverse} \text{ (detik)}$$

$$t_{T \text{ Total}} = \text{delay transmisi total (detik)}$$

3.10.1.3 Delay Propagasi

Delay Propagasi merupakan waktu perambatan atau penjaralan ke udara yang dibutuhkan oleh sebuah paket data. Besarnya nilai *delay* ini dihitung pada tiap *node* dengan lintasan *end to end* (MT, Node B, RNC, SGSN, GGSN, SGSN, RNC, Node B, MT) dimana antara MT dengan Node B menggunakan media udara dan antara Node B sampai dengan GGSN menggunakan media *fiber optic*. Persamaan *delay* propagasi secara umum ditentukan sesuai rumus berikut [Halsall, 1988:114] :

$$t_p = \frac{S}{V} \tag{3.58}$$

dengan :

$$t_p = \text{delay propagasi (detik)}$$

$$S = \text{jarak antara MT dengan Node B/radius sel (m)}$$

$$V = \text{kecepatan propagasi informasi pada media/saluran transmisi (meter/detik)}$$

Sedangkan untuk masing-masing lintasan adalah ditentukan dengan persamaan dibawah ini:

(a) Delay Propagasi dari MT ke Node B lintasan *forward*

$$t_{p1} = \frac{d}{c} \tag{3.59}$$

dengan :



t_{p1} = delay propagasi dari MT ke Node B saat lintasan *forward* (detik)

Nilai d pada hal ini adalah nilai *radius cell*. Pada skripsi ini nilai *radius cell* yang digunakan adalah untuk type sel *micro* dengan daerah operasi urban dan pusat kota sesuai dengan tabel 3.2 yaitu sebesar 0,4 km dengan kecepatan data maksimal 384 Kbps.

Sehingga besarnya *delay propagasi* ditentukan dengan persamaan:

$$t_{p1} = \frac{d}{c} = \frac{N_{frameMAC} \times R}{c} \quad (3.60)$$

dengan :

R = radius cell (m)

c = cepat rambat gelombang di udara 3×10^8 (m/detik)

(b) Delay Propagasi dari Node B ke RNC lintasan *forward*

Pada Node B dan RNC dihubungkan dengan STM1 dengan media transmisi adalah *fiber optic* dan diasumsikan jarak antara Node B ke RNC adalah 100 m.

Sehingga nilai *delay propagasinya* adalah:

$$t_{p2} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \quad (3.61)$$

dengan :

t_{p2} = *delay propagasi* dari Node B ke RNC saat lintasan *forward* (s)

d = jarak Node B - RNC (m)

c = cepat rambat gelombang di cahaya 2×10^8 (m/detik)

(c) Delay Propagasi dari RNC ke SGSN lintasan *forward*

Pada RNC ke SGSN dihubungkan dengan media transmisi *fiber optic* dengan jarak antara RNC ke SGSN diasumsikan adalah 25 m. Nilai *delay propagasi* dapat ditentukan dengan persamaan:

$$t_{p3} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \quad (3.62)$$

dengan :

t_{p3} = *delay propagasi* dari RNC ke SGSN saat lintasan *forward*

c = cepat rambat gelombang di cahaya 2×10^8 (m/detik)



(d) Delay Propagasi dari SGSN ke GGSN lintasan forward

Pada SGSN ke GGSN dihubungkan dengan media transmisi *fiber optic* dengan jarak antara SGSN ke GGSN diasumsikan adalah 25 m. Nilai *delay* propagasi antara SGSN ke GGSN didapatkan dengan menggunakan persamaan:

$$t_{p4} = \frac{d}{c} = \frac{N_{frameSGSN} \times d}{c} \quad (3.63)$$

dengan :

$$t_{p4} = \text{delay propagasi dari SGSN ke GGSN saat lintasan forward (detik)}$$

(e) Delay Propagasi dari GGSN ke SGSN lintasan reverse

Pada saat lintasan *reverse* dari GGSN ke SGSN sama dengan pada saat lintasan *forward* yaitu panjang 25 m dengan media transmisi *fiber optic*.

Sehingga *delay* propagasi sesuai dengan persamaan:

$$t_{p5} = \frac{d}{c} = \frac{N_{frameGGSN} \times d}{c} \quad (3.64)$$

dengan :

$$t_{p5} = \text{delay propagasi dari GGSN ke SGSN saat lintasan reverse (detik)}$$

(f) Delay Propagasi dari SGSN ke RNC saat lintasan reverse

Pada saat lintasan *reverse* dari SGSN ke RNC sama dengan pada saat lintasan *forward* yaitu panjang 25 m dengan media transmisi *fiber optic*. Sehingga *delay* propagasi adalah:

$$t_{p6} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \quad (3.65)$$

dengan :

$$t_{p6} = \text{delay propagasi dari SGSN ke RNC saat lintasan reverse (detik)}$$

(g) Delay Propagasi dari RNC ke Node B saat lintasan reverse

Pada saat lintasan *reverse* dari RNC ke Node B sama dengan pada saat lintasan *forward* yaitu panjang 100 m dengan media transmisi *fiber optic*. Untuk itu nilai *delay* propagasi ditentukan dengan persamaan:

$$t_{p7} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \quad (3.66)$$

dengan :

$$t_{p7} = \text{delay propagasi dari RNC ke Node B saat lintasan reverse (detik)}$$

(h) Delay Propagasi Node B ke MT lintasan *reverse*

Untuk *delay* propagasi saat lintasan *reverse* sama dengan saat lintasan *forward* dengan radius R sebesar 0,4 km. Nilai *delay* propagasi ini ditentukan dengan persamaan:

$$t_{p8} = \frac{d}{c} = \frac{N_{frameMAC} \times R}{c} \quad (3.67)$$

dengan :

$$t_{p8} = \text{delay propagasi dari Node B ke MT saat lintasan } reverse \text{ (detik)}$$

(i) Delay Propagasi total

Dari analisis diatas maka akan didapat nilai untuk *delay* propagasi untuk lintasan *forward*, lintasan *reverse*, dan *delay* propagasi total. Nilai *delay* propagasi total adalah dengan menjumlahkan seluruh nilai *delay* propagasi pada masing-masing lintasan atau jumlah dari *delay* untuk lintasan *forward* dan *reverse*.

Delay propagasi untuk lintasan *forward* ditentukan dengan persamaan :

$$t_{p \text{ forward}} = t_{p1} + t_{p2} + t_{p3} + t_{p4} \quad (3.68)$$

Delay propagasi untuk lintasan *reverse* ditentukan dengan persamaan :

$$t_{p \text{ reverse}} = t_{p5} + t_{p6} + t_{p7} + t_{p8} \quad (3.69)$$

Delay propagasi total ditentukan dengan persamaan :

$$t_{p \text{ Total}} = t_{p \text{ forward}} + t_{p \text{ reverse}} \quad (3.70)$$

dengan:

$$t_{p \text{ forward}} = \text{delay propagasi untuk lintasan } forward \text{ (detik)}$$

$$t_{p \text{ reverse}} = \text{delay propagasi untuk lintasan } reverse \text{ (detik)}$$

$$t_{p \text{ Total}} = \text{delay propagasi total (detik)}$$

3.10.1.4 Delay antrian

Delay antrian adalah waktu dimana paket data tersebut berada dalam antrian untuk ditransmisikan di GGSN. Selama waktu ini paket data menunggu sampai selesainya paket lain ditransmisikan. Lamanya waktu menunggu tiap paket untuk memperoleh akses tiap waktu berubah-ubah besarnya, namun jika antrian kosong dan tidak ada paket data lain yang sedang ditransmisikan maka *delay* antrian tidak terjadi atau sama dengan nol. *Delay* antrian yang dianalisis adalah *delay* yang terjadi pada GGSN dengan menggunakan model antrian M/M/1.

Model antrian M/M/1 berarti proses kedatangan paket data umumnya acak dan waktu pelayanan adalah distribusi eksponensial. M pertama menunjukkan distribusi kedatangan *Poisson*, M kedua berarti distribusi waktu pelayanan eksponensial, dan 1 menunjukkan bahwa jumlah *server* adalah tunggal. Disiplin antrian yang digunakan adalah FIFO (*First In First Out*). Adapun parameter model antrian M/M/1 adalah :

- Kapasitas *link* adalah C bps dengan menggunakan kecepatan standar GGSN dan panjang paket data adalah m bit. Besarnya kapasitas *link* akan menentukan kecepatan pelayanan. Sehingga kecepatan pelayanan di GGSN *server* :

$$\mu_{GGSN} = \frac{C_{GGSN}}{m} \tag{3.71}$$

dimana :

C_{GGSN} = kapasitas link pada GGSN (bps)

μ_{GGSN} = kecepatan pelayanan pada GGSN (paket / detik)

m = jumlah paket data pada GGSN (paket)

- Interval waktu untuk permintaan (*request*) merupakan distribusi *poisson* dengan kecepatan kedatangan data adalah λ paket/detik.

Besarnya total *delay* antrian adalah :

$$t_w = t_q + t_s \tag{3.72}$$

dimana :

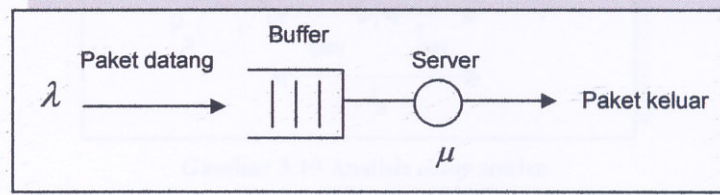
t_q = waktu tunggu paket dalam saluran (detik)

t_s = rata-rata kecepatan pelayanan paket pada GGSN (detik)

dimana

$$t_s = \frac{1}{\mu_{GGSN}} \tag{3.73}$$





Gambar 3.9 Model antrian *single server*
 Sumber : Schwartz, 1987:22

Untuk performansi sistem antrian, ditunjukkan dalam bentuk ρ (*utilization*), yang nilainya diasumsikan berubah dengan kenaikan tertentu.

$$\rho = \frac{\lambda}{\mu} \tag{3.74}$$

dimana :

- ρ = faktor utilisasi ($0 < \rho < 1$)
- λ = laju kedatangan paket pada server (paket/detik)
- μ = kecepatan pelayanan pada server (paket/detik)

Sehingga untuk masing-masing nilai ρ dapat ditentukan laju kedatangan paket (λ) GGSN sebesar :

$$\lambda_{\text{GGSN}} = \rho \times \mu_{\text{GGSN}}$$

kemudian dengan menggunakan teori *little*, besarnya nilai waktu antrian didapatkan :

$$t_w = \frac{1}{\mu(1-\rho)} \tag{3.75}$$

dimana :

$$t_w = \text{delay antrian (detik)}$$

Sedangkan waktu tunggu dari paket data dapat dirumuskan :

$$\begin{aligned} t_q &= t_w + t_s \\ &= \frac{1}{\mu} - \frac{1}{\mu} + \frac{\lambda}{\mu^2} \\ &= \frac{\lambda}{\mu} = \frac{\lambda}{\mu - \lambda} \end{aligned}$$

Sehingga total *delay* antrian pada GGSN adalah :



$$t_w = t_q + t_s = \frac{\lambda_{GGSN} / \mu_{GGSN}}{\mu_{GGSN} - \lambda_{GGSN}} + \frac{1}{\mu_{GGSN}} \quad (3.76)$$

3.10.2 Probabilitas Packet Loss Jaringan UMTS

Seperti pada persamaan 2.14 bahwa untuk menghitung probabilitas *packet loss* total aplikasi *videophone* yaitu :

$$\rho_{tot} = 1 - [(1 - \rho_{Network})(1 - \rho_{UDP/RTP/IP})]$$

dimana probabilitas *packet loss* pada jaringan UMTS ($\rho_{Network}$) dihitung dari probabilitas *packet loss* yang terjadi pada *air interface* (WCDMA *air interface*) sistem UMTS, UTRAN (Node B, RNC) serta pada *core network* (SGSN, GGSN) dengan persamaan:

$$\rho_{network} = 1 - [(1 - \rho_{AI})(1 - \rho_{UTRAN})^{h_1}(1 - \rho_{corenetwork})^{h_2}] \quad (3.77)$$

dimana :

- $\rho_{network}$ = probabilitas *packet loss* pada jaringan UMTS
- ρ_{AI} = probabilitas *packet loss* pada *air interface* (WCDMA *Air interface*)
- ρ_{UTRAN} = probabilitas *packet loss* pada UTRAN (Node B, RNC)
- $\rho_{core network}$ = probabilitas *packet loss* pada *core network* (SGSN, GGSN)
- h_1 = jumlah hop pada jaringan UTRAN (Node B, RNC)
- h_2 = jumlah hop pada jaringan *core network* (SGSN, GGSN)

Probabilitas *packet loss* pada UTRAN dan *core network* dihitung dari probabilitas bit *error* (BER) di jaringan tersebut, dengan persamaan :

$$\rho_{UTRAN} = (l + l')Pe \quad (3.78)$$

dimana :

- l = panjang paket (byte)
- l' = panjang header IPv6 (byte)
- Pe = probabilitas bit *error* (BER) (10^{-7})

$$\rho_{corenetwork} = (l + l')Pe \quad (3.79)$$

dimana :

- l = panjang paket (byte)
- l' = panjang header IPv6 (byte)



$P_e =$ probabilitas bit error (BER) (10^{-7})

Untuk menghitung probabilitas packet loss pada *air interface* (WCDMA *air interface*) baik untuk lintasan *forward* maupun lintasan *reverse* diperlukan parameter-parameter *link budget* pada tabel 3.4.

a. Lintasan Forward (MS ke BTS)

➤ *EIRP (Effective Isotropic Radiated Power) mobile station*

Merupakan penjumlahan daya *output mobile station*, rugi-rugi sistem transmisi dan *gain* antena *mobile station* pada arah radiasi maksimum :

$$EIRP_{MS} = P_{TX-MS} + G_{MS} - L_{MS} \tag{3.80}$$

dimana :

$EIRP_{MS}$ = *EIRP mobile station* (dBm)

P_{TX-MS} = Daya transmisi *mobile station* (dBm)

G_{MS} = *gain* antena *mobile station* (dBi)

L_{MS} = Rugi kabel, konektor, dan kombiner (dB)

➤ *Interference Margin*

Merupakan fungsi dari jumlah pembebanan pada *carrier* WCDMA.

$$I_m = 10 \log \left(\frac{1}{1-\eta} \right) \tag{3.81}$$

dimana :

I_m = *interference margin* (dB)

η = faktor pembebanan sistem (75 %)

➤ Daya penerimaan *base station* tiap *user*

$$P_{cu} = EIRP_{MS} - L_p + G_{BS} + L_f - I_m \tag{3.82}$$

dimana :

P_{cu} = daya penerimaan *base station* tiap *user* (dBm)

L_p = rugi-rugi propagasi (dB)

G_{BS} = *gain* antena *base station* (dBi)

L_f = *loss feeder* (dB)

➤ *Thermal noise*

$$N_o = 10 \log (290 \times 1,38 \cdot 10^{-23}) + NF_{BS} + 30 \tag{3.83}$$

dimana :

N_o = *thermal noise* (dBm/Hz)



NF_{BS} = noise figure base station (dB)

- Energy per bit to noise density ratio

$$E_b/N_o = P_{cu} - 10 \log BR - 10 \log 10^{0,1N_o} \quad (3.84)$$

dimana :

E_b/N_o = signal to noise ratio (dB)

BR = data rate (bps)

b. Lintasan Reverse (BTS ke MS)

- Daya radiasi efektif kanal trafik

$$p_t = P_t - 10 \log N_t - 10 \log C_f \quad (3.85)$$

p_t = ERP kanal trafik (dBm)

P_t = ERP seluruh kanal trafik dari base station (dBm)

N_t = jumlah kanal trafik pada sektor sel

C_f = faktor aktivitas kanal suara/data (berkisar antara 0,4 – 0,6)

- Daya yang diterima kanal trafik

$$p_{tr} = p_t - L_p + G_{MS} + L_{MS} - I_m \quad (3.86)$$

p_{tr} = daya yang diterima kanal trafik (dBm)

- Thermal noise

$$N_o = 10 \log (290 \times 1,38 \cdot 10^{-23}) + NF_{MS} + 30 \quad (3.87)$$

dimana :

N_o = thermal noise (dBm/Hz)

NF_{MS} = noise figure mobile station (dB)

- Energy per bit to noise density ratio kanal trafik

$$E_b/N_o = p_{tr} - 10 \log BR - 10 \log 10^{0,1N_o} \quad (3.88)$$

dimana :

E_b/N_o = signal to noise ratio (dB)

BR = data rate (bps)

Sehingga BER (P_e) secara umum dapat didefinisikan [Pritchard, et.al.

1993:338]:

$$P_e = \frac{e^{-E_b/N_o}}{2\sqrt{\pi}(E_b/N_o)} \quad (3.89)$$

Dan probabilitas packet loss pada air interface jaringan UMTS baik untuk lintasan forward maupun lintasan reverse dihitung dengan persamaan :

$$\rho_{AI} = 1 - (1 - Pe)^{l+l'} \quad (3.90)$$

dengan :

- l = panjang data (byte)
- l' = panjang *header* IPv6 (byte)
- Pe = probabilitas bit *error* (BER)

Tabel 3.4 Parameter *Link Budget* W-CDMA

Parameter Link Budget W-CDMA	
<i>Mobile Station (MS)</i>	
Tinggi Antena <i>Mobile Station</i>	1,5 m
Daya Kirim MS P_{MS-TX}	23 dBm
<i>MSsensitivity</i>	-107 dBm
Gain Antena <i>Mobile Station</i> (G_{MS})	0 dB
<i>Base Station (BS)</i>	
Frekuensi (fc)	2000 MHz
Tinggi Antena (h_{BS})	40 m
<i>ERP base station</i> (ERP_{BS})	43 dBm
Jumlah kanal/sector	76 (Urban), 37(Sub Urban)
Faktor aktifitas kanal trafik	0,60
Gain Antena BS (G_{BS})	18 dB
<i>Loss Feeder RF Base Station</i>	2 dB
<i>Diversity Gain BS</i> (G_{div})	3 dB
<i>Fade Margin</i> (Fm)	4 dB
<i>Gain Soft Handoff</i> (G_{SHO})	3 dB
Pembebanan atau <i>load</i> efektif (η)	10%-95%
Rugi propagasi rata-rata (L_p)	138 dB
<i>Base Station Sensitivity</i>	-113 dBm
<i>Noise figure base station</i>	7 dB
<i>Noise figure mobile station</i>	10 dB
<i>Bit rate</i> kanal trafik arah <i>forward</i>	384000 bps
<i>Bit rate</i> kanal trafik arah <i>reverse</i>	384000 bps

Sumber : NTIA, 2000

3.10.3 Throughput

Throughput merupakan parameter yang digunakan untuk mengetahui jumlah data yang diterima dalam keadaan baik terhadap waktu total transmisi yang dibutuhkan dari sumber data ke penerima. Pengiriman data pada jaringan *packet switch* antara 2 stasiun yang melalui beberapa lapisan protokol mempunyai batas kemampuan kapasitas yang bisa dikeluarkan oleh jaringan tersebut. Hal ini

dikenal dengan istilah *throughput*. Dalam skripsi ini *throughput* dihitung pada lintasan MT (*mobile terminal*) pengirim hingga MT (*mobile terminal*) penerima. Besarnya nilai *throughput* jaringan dapat dirumuskan [Schwartz, 1987:131]:

$$\lambda = \frac{\text{jumlah paket data benar yang diterima}}{t_v} \quad (3.91)$$

dengan:

$$\begin{aligned} \lambda &= \text{throughput (paket / detik)} \\ t_v &= \text{waktu total transmisi untuk mengirimkan paket yang benar (detik)} \end{aligned}$$

Throughput yang didapat dengan memperhitungkan probabilitas *packet loss* (ρ) adalah :

$$\lambda = \frac{1}{t_v} = \frac{(1 - \rho_{tot})}{t_1 [1 + (\alpha - 1) \rho_{tot}]} \quad (3.92)$$

dengan :

$$\begin{aligned} \lambda &= \text{throughput (paket/s)} \\ t_1 &= \text{waktu transmisi sebuah paket (detik)} \\ \alpha &= \text{konstanta} \\ \rho_{tot} &= \text{probabilitas packet loss total / end- to- end} \\ t_v &= \text{waktu transmisi rata-rata sebuah paket yang diterima dalam keadaan benar (detik)} \end{aligned}$$

Sedangkan waktu transmisi paket [Schwartz, 1987 : 132] ditentukan dengan persamaan :

$$t_1 = \frac{(PL_{paket} + H_{paket}) \times 8}{C_{trans}} \quad (3.93)$$

dengan:

$$\begin{aligned} t_1 &= \text{waktu yang dibutuhkan untuk mentransmisikan sebuah paket melalui jaringan UMTS (detik)} \\ PL_{paket} &= \text{payload paket PDU UMTS (byte)} \\ H_{paket} &= \text{panjang header paket PDU UMTS (byte)} \\ C_{trans} &= \text{kapasitas saluran transmisi (bps)} \end{aligned}$$

Dengan mengetahui nilai t_T dan t_I dapat dihitung parameter α [Schwartz, 1987 : 126] :

$$\alpha = \frac{t_T}{t_I} \tag{3.94}$$

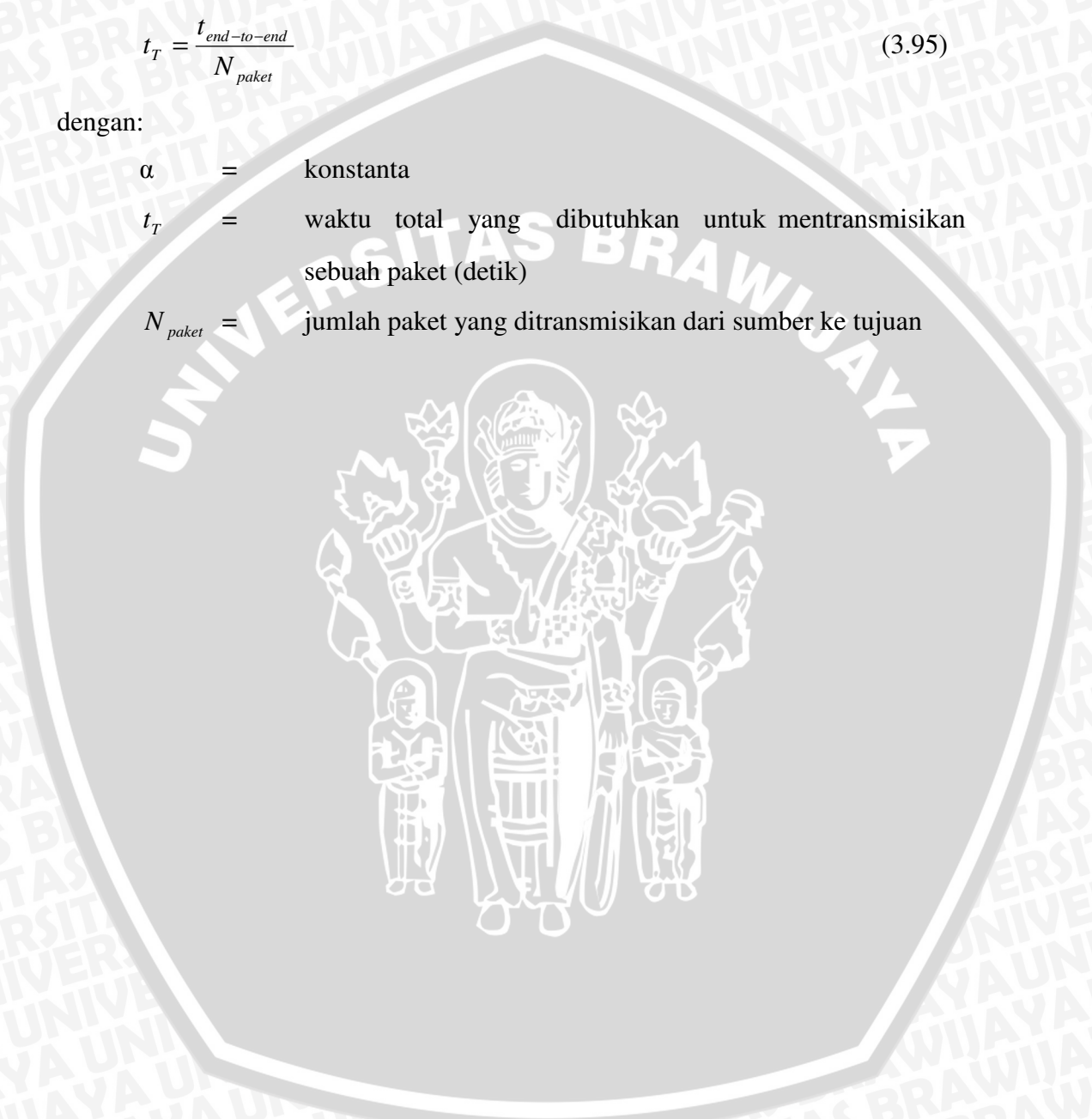
$$t_T = \frac{t_{\text{end-to-end}}}{N_{\text{paket}}} \tag{3.95}$$

dengan:

α = konstanta

t_T = waktu total yang dibutuhkan untuk mentransmisikan sebuah paket (detik)

N_{paket} = jumlah paket yang ditransmisikan dari sumber ke tujuan



BAB IV

METODOLOGI PENELITIAN

Kajian yang digunakan dalam skripsi ini adalah kajian yang bersifat analisis, yaitu Analisis Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS dalam bentuk konsep yang mengacu pada studi kepustakaan. Data yang digunakan berupa data sekunder yang didapat dari dari berbagai buku teks, jurnal-jurnal, dan *download* dari *internet*. Metode yang digunakan dalam skripsi ini adalah metode pengambilan data, metode pembatasan variabel, dan metode analisis data.

4.1 Metode Pengambilan Data

Data yang digunakan dalam kajian ini berupa data sekunder, yaitu data yang diperoleh dari buku referensi, jurnal, skripsi dan standar dari ITU (*International Telecommunication Union*) dan ETSI (*European Telecommunication Standard System*). Adapun data sekunder yang digunakan dalam pembahasan skripsi ini antara lain :

4.1.1 Spesifikasi Internet Protokol versi 6 (IPv6)

Data ini merupakan data pokok yang diperlukan untuk analisis penerapan IPv6 sebagai komunikasi data untuk aplikasi *videophone* pada jaringan UMTS. Data spesifikasi IPv6 ini diperlukan untuk menganalisis proses aliran data pada tiap-tiap *layer*. Data-data yang digunakan dalam kajian ini meliputi :

- Konfigurasi protokol TCP/IP.
- Format *header* IPv6.

4.1.2 Spesifikasi Aplikasi *Videophone*

Data ini merupakan data pokok yang diperlukan untuk analisis penerapan aplikasi *videophone* pada jaringan UMTS. Data spesifikasi aplikasi *videophone* ini juga diperlukan untuk menentukan nilai *delay* aplikasi *videophone*, *delay gatekeeper*, serta *packet loss* pada aplikasi *videophone*. Data-data yang digunakan dalam kajian ini meliputi :



- Konfigurasi jaringan aplikasi *videophone*.
- Protokol aplikasi *videophone*.
- *Bandwith CODEC, delay CODEC, serta frame rate.*

4.1.3 Spesifikasi Jaringan UMTS

Data ini merupakan data pokok yang diperlukan untuk menentukan model serta ukuran jaringan *mobile cellular* UMTS. Data spesifikasi jaringan ini juga diperlukan untuk menganalisis parameter jaringan yang meliputi *delay* jaringan, *packet loss*, serta *throughput* sehingga dapat memenuhi kualitas pelayanan data.

Data-data yang digunakan dalam kajian ini meliputi :

- Konfigurasi jaringan UMTS.
- Protokol jaringan UMTS berbasis IPv6.
- Data spesifikasi teknis jaringan UMTS.

4.2 Metode Pembatasan Variabel

Metode pembatasan variabel digunakan untuk membatasi permasalahan yang disampaikan sehingga tidak melebihi ataupun kurang dari format fokus yang ingin disampaikan. Beberapa batasan untuk variabel data ini antara lain :

- Sistem UMTS yang digunakan adalah menggunakan frekuensi operasi 1920-1980 MHz (*uplink*) dan 2110-2170 MHz (*downlink*) dengan lebar kanal 5 MHz dan spasi 200 KHz, dengan metode akses yang digunakan yaitu FDD W-CDMA.
- *Radius cell* yang digunakan adalah untuk type sel *micro* dengan daerah operasi urban dan pusat yaitu sebesar 0,4 km dengan kecepatan data maksimal 384 Kbps.
- Model hubungan *end-to-end* yang digunakan adalah antar MT dalam GGSN yang sama dengan koneksi dari MT pengirim ke MT penerima yaitu MT-UTRAN(Node B dan RNC)-SGSN-GGSN-SGSN-UTRAN(Node B dan RNC)-MT.
- Aplikasi *videophone* yang diterapkan pada jaringan UMTS merupakan aplikasi multimedia dengan standar ITU-T H.323.
- Jenis *CODEC* yang digunakan untuk aplikasi *videophone* adalah H.263 untuk *video* dan untuk *audio* adalah G.723.1.

- Protokol yang digunakan aplikasi *videophone* adalah protokol RTP, UDP, IPv6 dengan besar *header* 60 byte.
- *Bandwith CODEC audio* yang digunakan sebesar 6,3 kbps serta *bandwith CODEC video* berkisar 10 – 384 kbps dengan format gambar CIF (*Common Intermediate Format*) menggunakan *frame rate* 30 ms.

4.3 Metode Analisis Data

Metode analisis data yang digunakan dalam penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS ini adalah analisis matematis yaitu dengan mengumpulkan nilai dari parameter-parameter data sekunder dengan kesesuaian terhadap standar yang digunakan, untuk kemudian dianalisis berdasarkan pembahasan yang telah diuraikan. Analisis yang dilakukan meliputi :

- Perhitungan panjang paket data (*payload*) *videophone*.
- Perhitungan *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.
- Perhitungan *packet loss* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.
- Perhitungan *throughput* total.

4.4 Menarik Kesimpulan

Dari hasil analisis sistem, maka akan dapat ditarik kesimpulan mengenai penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS.

BAB V

ANALISIS PENERAPAN IPv6 UNTUK APLIKASI VIDEOPHONE PADA JARINGAN UMTS

Penerapan IPv6 untuk aplikasi *videophone* ini berdasar pada aplikasi yang dapat dijalankan pada jaringan UMTS. Untuk itu ada beberapa hal yang harus diperhatikan agar penerapan IPv6 untuk aplikasi *videophone* dapat diterapkan pada jaringan UMTS, yaitu :

- Alokasi *bandwidth* aplikasi yang berhubungan dengan QoS layanan.
- Perhitungan *delay* yang berhubungan dengan QoS aplikasi *videophone* dan jaringan UMTS.

5.1 Paket Data Aplikasi *Videophone*

Berdasarkan tabel 2.1, aplikasi *videophone* menggunakan jenis *CODEC* H.263 untuk *video* dengan *bandwidth CODEC* antara 10 - 384 kbps dan G.723.1 untuk *audio* atau suara dengan *bandwidth* 6,3 kbps. Dan juga format gambar yang digunakan berdasarkan Tabel 2.2 yaitu CIF dengan *frame rate* 30 ms. Dengan menggunakan persamaan 2.1 dan 2.2 maka besar *payload* paket data aplikasi *videophone* dan konsumsi *bandwidth* aktual aplikasi *videophone* pada jaringan UMTS dapat dihitung :

$$PLa = B_{CODECA} \times frame\ rate = (6,3 \cdot 10^3) \times (30 \cdot 10^{-3}) = 189 \text{ bit}$$

$$PLV = B_{CODECV} \times frame\ rate = (64 \cdot 10^3) \times (30 \cdot 10^{-3}) = 1920 \text{ bit}$$

Sehingga besar paket data aplikasi *videophone* pada jaringan UMTS berdasarkan persamaan 2.5 adalah sebesar :

$$\begin{aligned} Pvp\text{-size} &= header_{UDP/RTP/IPv6} + (PLa + PLV) = 480 + (189+1920) \\ &= 2589 \text{ bit} \\ &= 323,625 \text{ byte} \end{aligned}$$

Dan dengan menggunakan persamaan 2.3 sampai 2.8 maka besar konsumsi *bandwidth* aktual aplikasi *videophone* pada jaringan UMTS dapat dihitung :

$$Pa\text{-size} = header_{UDP/RTP/IPv6} + PLa = 480 + 189 = 669 \text{ bit}$$

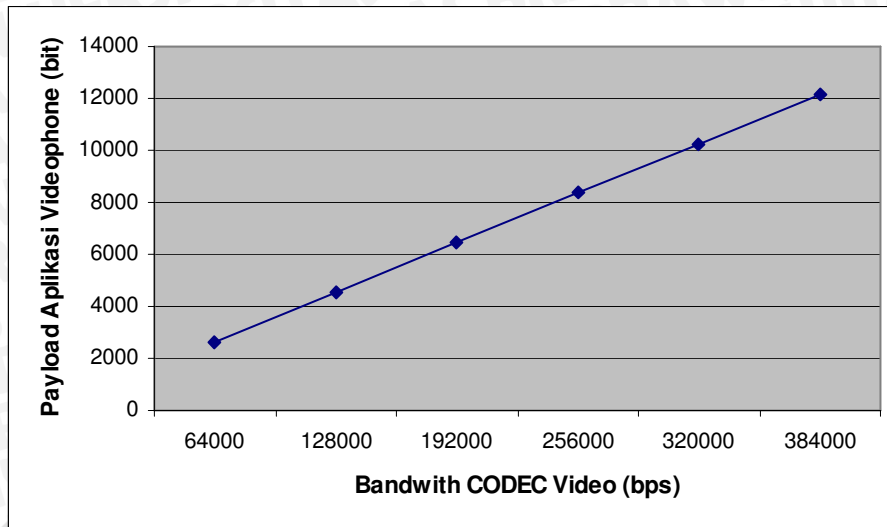
$$\begin{aligned}
 PV\text{-size} &= \text{header}_{\text{UDP/RTP/IPv6}} + \text{PLV} = 480 + 1920 = 2400 \text{ bit} \\
 Pv &= B_{\text{CODECV}} / \text{PLV} = 64000/1920 = 33,33 \text{ paket per second} \\
 Pa &= B_{\text{CODECA}} / \text{PLa} = 6300/189 = 33,33 \text{ paket per second} \\
 Bv &= PV\text{-size} \times Pv = 2400 \times 33,33 = 80000 \text{ bit per second (bps)} \\
 Ba &= Pa\text{-size} \times Pa = 669 \times 33,33 = 22300 \text{ bit per second (bps)} \\
 Bvp &= Bv + Ba + \text{bandwith overhead} \\
 &= 80000 + 22300 + \{5,2 \% \times (80000 + 22300)\} \\
 &= 102300 + 5319,6 \\
 &= 107620 \text{ bps}
 \end{aligned}$$

Dengan cara yang sama, hasil analisis besar paket data aplikasi *videophone* (PLvp-size) dan konsumsi *bandwidth* aktual aplikasi *videophone* berbasis IPv6 pada jaringan UMTS untuk *bandwidth CODEC video* sebesar 64 kbps, 128 kbps, 192 kbps, 256 kbps, 320 kbps dan 384 kbps ditunjukkan pada Tabel 5.1.

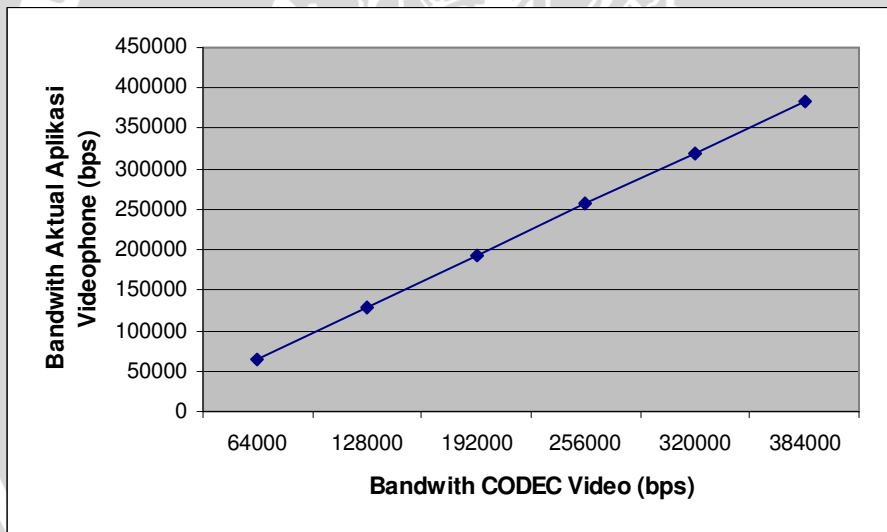
Tabel 5.1 Hasil analisis paket data aplikasi *videophone* dan *bandwidth* aplikasi *videophone* berbasis IPv6 pada jaringan UMTS

No.	<i>Bandwidth CODEC Video</i>	<i>Panjang Paket Videophone</i>	<i>Bandwidth Aktual Aplikasi Videophone</i>
1.	64 kbps	2589 bit	107620 bps
2.	128 kbps	4509 bit	174948 bps
3.	192 kbps	6429 bit	242276 bps
4.	256 kbps	8349 bit	309604 bps
5.	320 kbps	10269 bit	376932 bps
6.	384 kbps	12189 bit	444260 bps

Sumber : Perhitungan



Gambar 5.1 Grafik hubungan *bandwith CODEC video* dengan besarnya paket data (*payload*) aplikasi *videophone* berbasis IPv6



Gambar 5.2 Grafik hubungan *bandwith CODEC video* dengan *bandwith* aktual aplikasi *videophone* berbasis IPv6

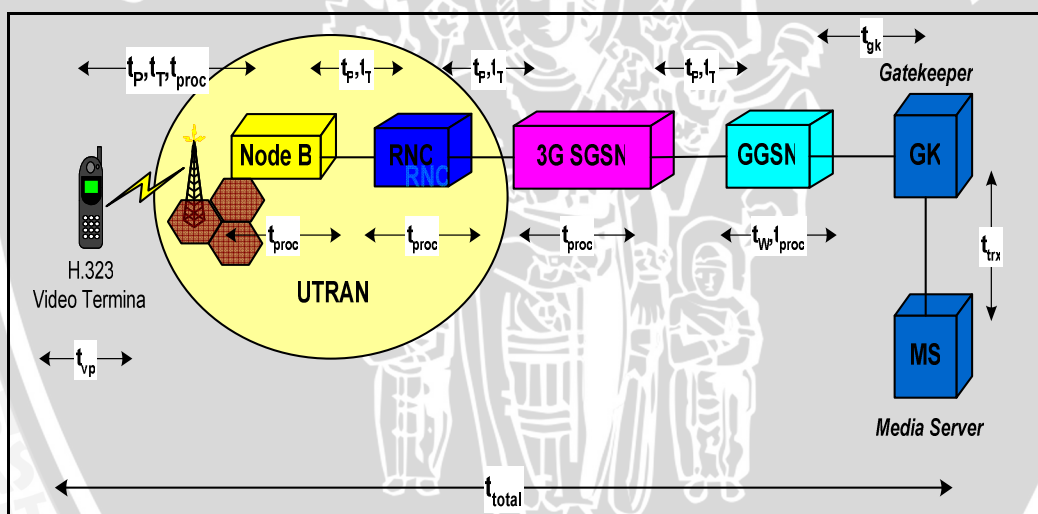
Dari hasil perhitungan dan grafik di atas dapat disimpulkan bahwa :

- Semakin besar *bandwidth CODEC video* yang digunakan maka paket *videophone* yang dihasilkan juga semakin besar.
- Semakin besar *bandwidth CODEC video* yang digunakan maka *bandwidth* aktual aplikasi *videophone* berbasis IPv6 yang dibutuhkan juga semakin besar.

- Paket *videophone* terbesar yaitu 12189 bit (1523,625 byte) dengan menggunakan *bandwidth CODEC video* sebesar 384 kbps. Sedangkan paket *videophone* terkecil adalah 2589 bit (323,625 byte) dengan menggunakan *bandwidth CODEC video* 64 kbps.
- *Bandwidth* aplikasi *videophone* berbasis IPv6 terbesar adalah 444260 bps dengan *bandwidth CODEC video* sebesar 384 kbps. Sedangkan yang terkecil adalah 107260 bps dengan *bandwidth CODEC video* 64 kbps.

5.2 Delay End-to-End Penerapan IPv6 untuk Aplikasi Videophone pada Jaringan UMTS

Model hubungan yang digunakan dalam analisis *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS adalah model hubungan antara MT pengirim dengan MT penerima seperti pada gambar 3.5 dan ditunjukkan dengan model blok diagram seperti pada gambar 5.3 dibawah ini.



Gambar 5.3 Analisis *delay end-to-end*
 Sumber : Analisis

5.2.1 Delay Aplikasi Videophone

Dalam penerapan aplikasi *videophone* pada jaringan UMTS membutuhkan beberapa perangkat untuk menjadikan aplikasi ini lebih interaktif bagi *user* seperti kamera *video*, layar *video*, sistem pemrosesan suara (*audio*) dan gambar (*video*) Seperti yang pada gambar 2.12, seluruh perangkat tersebut terintegrasi menjadi satu pada MT (*Mobile Terminal*) sebagai perangkat *videophone*. Selain itu terdapat juga *media server* (Gambar 3.5) sebagai penyedia layanan *videophone*.

Delay aplikasi *videophone* adalah waktu yang dibutuhkan dalam pemrosesan sinyal *audio* dan *video* hingga menjadi paket *videophone* yang siap untuk ditransmisikan pada jaringan UMTS. Sehingga *delay* aplikasi *videophone* merupakan *delay encoding* dan *decoding* yang disamakan besarnya waktu yang dibutuhkan untuk menghasilkan satu paket *videophone* yaitu 30 ms, besarnya *delay* aplikasi *videophone* dihitung dengan persamaan 2.9 :

$$t_{CODEC} = t_a + t_v = 30 \text{ ms} + 30 \text{ ms} = 60 \text{ ms}$$

5.2.2 Delay Gatekeeper

Dalam hal ini *delay gatekeeper* merupakan *delay* yang dialami paket selama berada pada sistem *gatekeeper*, sebelum dikirim atau diterima dari dan ke jaringan UMTS sebagaimana yang telah dijelaskan pada bab 2. Dengan menggunakan kecepatan standar *gatekeeper* sebesar 100 Mbps dan panjang data di *gatekeeper* sebesar 1500 byte serta panjang data pada *media server* sebesar 1518 byte (Subchi, 2005 :76). Waktu rata-rata setiap paket data berada dalam sistem *gatekeeper* dihitung dengan persamaan 2.10 – 2.12 :

$$\mu_{gk} = \frac{C}{P} \times 8 = \frac{100 \cdot 10^6}{1500 \times 8} = 8333,33 \text{ paket/s}$$

$$T_{trx} = \frac{P}{C} + \text{interpacket gap} = \frac{1500 \times 8}{100 \cdot 10^6} + 3 \cdot 10^{-6} = 1,23 \cdot 10^{-4} \text{ s}$$

$$\lambda_{gk} = \frac{m}{T_{trx}} = \frac{1518 \times 8}{1,23 \cdot 10^{-4}} = 98,73 \cdot 10^6 \text{ paket/s}$$

$$t_{gk} = \frac{1}{\lambda_{gk} - \mu_{gk}} = \frac{1}{(98,73 \cdot 10^6) - (8333,33)} = 1,0129 \cdot 10^{-8} \text{ s}$$

5.2.3 Delay Jaringan UMTS

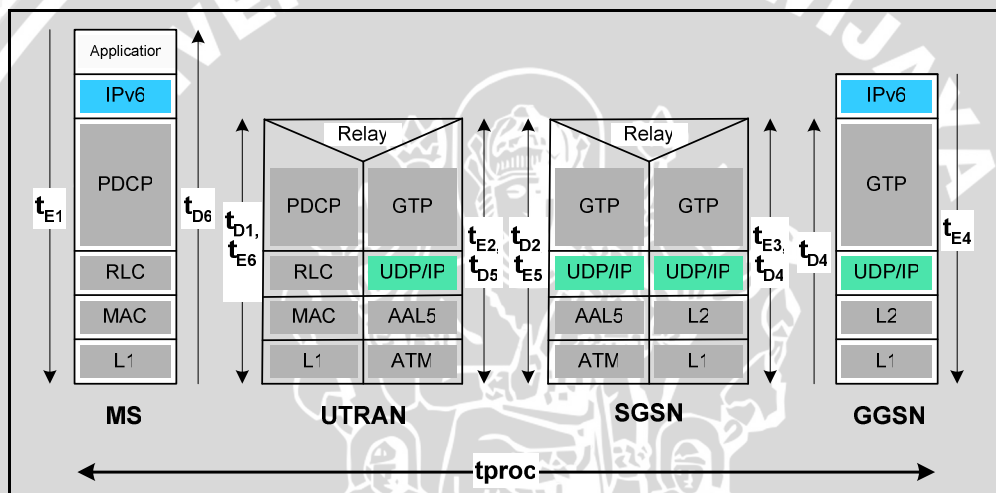
Salah satu ukuran unjuk kerja adalah *delay* yang dibutuhkan untuk mengirimkan data dari sumber ke tujuannya. Definisi umum dari *delay* adalah waktu tunda. Dalam jaringan berbasis *packet switching*, *delay* yang terjadi merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari sumber ke tujuannya pada setiap *hop*.

Panjang *segment* yang digunakan sebagai bahan analisis adalah panjang paket *videophone* dengan *bandwidth CODEC video* sebesar 64 kbps, 128 kbps,

192 kbps, 256 kbps, 320 kbps dan 384 kbps. Sedangkan *data rate* UMTS yang digunakan adalah 384 Kbps.

5.2.3.1 Delay Proses

Delay proses pada jaringan UMTS berupa *delay* enkapsulasi dan *delay* dekapsulasi. Perhitungan *delay* enkapsulasi dan dekapsulasi data melalui jaringan UMTS berbasis IPv6 berlaku untuk masing-masing *layer* seperti dalam gambar 5.4, yaitu dengan cara menambahkan *header* dan perubahan format data mulai dari MT (*Mobile Terminal*) / MS (*Mobile Station*) sampai GGSN (*Gateway GPRS Support Node*) untuk lintasan *forward* dan dari GGSN sampai MT untuk lintasan *reverse*.



Gambar 5.4 Analisis *delay* proses pada jaringan UMTS
sumber : Analisis

Adapun *delay* enkapsulasi/dekapsulasi untuk masing-masing *node* sesuai dengan model jaringan adalah :

➤ *Mobile Terminal (Forward)*

Sebagai contoh perhitungan digunakan *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps dan 1523,625 untuk *bandwidth CODEC video* 384 Kbps. *Segment* data aplikasi *videophone* dikirimkan ke *transport layer*, karena *videophone* merupakan aplikasi *realtime* maka pada *transport layer* ditambah *header* UDP sesuai persamaan 3.2 :

$$\begin{aligned} W_{\text{segmen}} &= W_{\text{data}} + \text{Header UDP/TCP} \\ &= 323,625 \text{ byte} + 8 \text{ byte} \end{aligned}$$

$$= 331,625 \text{ byte}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh 1531,625 byte.

Dari *layer transport* kemudian data dibawa menuju *network layer*, yaitu IP untuk diubah menjadi *datagram* dengan penambahan *header IPv6* 40 byte sesuai persamaan 3.3:

$$\begin{aligned} W_{\text{datagram}} &= W_{\text{segmen}} + \text{Header}_{\text{IPv6}} + \text{Header ICMP} \\ &= 331,625 \text{ byte} + 40 \text{ byte} + 8 \text{ byte} \\ &= 379,625 \text{ byte} \end{aligned}$$

Dan untuk *segment* data 1523,625 byte diperoleh 1579,625 byte.

Pada *link layer* PDCP, *datagram* diubah menjadi PDCP Data PDU. Dengan menggunakan persamaan 3.4 diperoleh panjang :

$$\begin{aligned} W_{\text{PDCP}} &= W_{\text{datagram}} + \text{Header}_{\text{PDCP}} \\ &= 379,625 \text{ byte} + 1 \text{ byte} \\ &= 380,625 \text{ byte} \end{aligned}$$

Untuk *segment* data 1523,625 byte karena panjang *datagram* IP melebihi MTU PDCP, maka *datagram* IP akan segmentasi. Dengan menggunakan persamaan 3.5 diperoleh :

$$N_{\text{frame-PDCP}} = \frac{W_{\text{datagram}}}{\text{MTU}_{\text{PDCP}}} = \frac{1579,625 \text{ byte}}{1500 \text{ byte}} = 1,053 \text{ buah}$$

Kemudian dengan persamaan 3.6 diperoleh panjang *frame* PDCP sebesar :

$$\begin{aligned} W_{\text{frame PDCP}} &= \text{MTU}_{\text{PDCP}} + \text{Header}_{\text{PDCP}} \\ &= 1500 \text{ byte} + 1 \text{ byte} \\ &= 1501 \text{ byte} \end{aligned}$$

Sehingga jumlah total *frame* PDCP yang dikirimkan ke RLC sesuai persamaan 3.7 adalah :

$$\begin{aligned} W_{\text{frame PDCP total}} &= N_{\text{frame PDCP}} \times W_{\text{frame PDCP}} \\ &= 1,053 \times 1501 \text{ byte} \\ &= 1580,678 \text{ byte} \end{aligned}$$

Pada *layer* RLC, PDCP Data PDU disegmentasi menjadi RLC PDU *fixed size* sebesar 40 byte, dengan persamaan 3.8 diperoleh :

$$N_{frame-RLC} = \frac{W_{PDCP}}{40byte} = \frac{380,625byte}{40byte} = 9,515 \text{ buah}$$

Kemudian dengan persamaan 3.9 diperoleh panjang *frame* RLC sebesar :

$$\begin{aligned} W_{frame RLC} &= 40 \text{ byte} + \text{header RLC} \\ &= 40 \text{ byte} + 2 \text{ byte} \\ &= 42 \text{ byte} \end{aligned}$$

Sehingga jumlah total *frame* RLC yang dikirimkan ke MAC adalah :

$$\begin{aligned} W_{total frame RLC} &= N_{frame RLC} \times W_{frame RLC} \\ &= 9,515 \times 42 \text{ byte} \\ &= 399,656 \text{ byte} \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 1659,711 byte

Pada *layer* MAC, untuk menjadi paket data yang siap untuk ditransmisikan lewat *physical layer* maka panjang *frame* RLC PDU akan ditambah *header* MAC sebesar 3 byte dan CRC 2 byte sesuai dengan persamaan 3.11 diperoleh :

$$\begin{aligned} W_{frame MAC} &= \text{Header MAC} + W_{frame RLC} + \text{CRC} \\ &= 3 \text{ byte} + 42 \text{ byte} + 3 \text{ byte} \\ &= 48 \text{ byte} \end{aligned}$$

Sehingga jumlah total *frame* MAC PDU yang siap untuk ditansmisikan lewat *physical layer* dengan persamaan 3.12 adalah :

$$\begin{aligned} W_{total frame MAC} &= N_{frame RLC} \times W_{frame MAC} \\ &= 9,515 \times 48 \text{ byte} \\ &= 456,75 \text{ byte} \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 1896,813 byte.

Nilai *delay* enkapsulasi pada MT untuk *segment* data 323,625 byte saat lintasan *forward* sesuai dengan persamaan 3.13 yaitu :

$$\begin{aligned} t_{E1} &= \frac{W_{total fraame MAC} \times 8}{V_{MT}} \\ &= \frac{456,75 \times 8}{384 \times 10^3} = \frac{3654bit}{384 \times 10^3 bps} = 9,5 \times 10^{-3} s \end{aligned}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi yang terjadi pada MT saat lintasan *forward* untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.2

Tabel 5.2 Hasil analisis *delay* enkapsulasi pada MT saat lintasan *forward*

No.	Panjang Paket Videophone	Delay Enkapsulasi
1.	323,625 byte	$9,515 \cdot 10^{-3}$ s
2.	563,625 byte	$1,551 \cdot 10^{-2}$ s
3.	803,625 byte	$2,151 \cdot 10^{-2}$ s
4.	1043,625 byte	$2,751 \cdot 10^{-2}$ s
5.	1283,625 byte	$3,351 \cdot 10^{-2}$ s
6.	1523,625 byte	$3,951 \cdot 10^{-2}$ s

Sumber : Perhitungan

➤ **UTRAN (Forward)**

Pada UTRAN menggunakan *interface* STM-1 dengan kecepatan transmisi adalah 155 Mbps. Nilai *delay* dekapsulasi pada UTRAN untuk *segment* data 323,625 byte saat lintasan *forward* dapat dihitung dengan persamaan 3.14 yaitu:

$$t_{D1} = \frac{W_{totalframeMAC}}{V_{UTRAN}} = \frac{3654bit}{155 \times 10^6 bps} = 2,357 \times 10^{-5} s$$

Pada UTRAN *frame* MAC PDU disegmentasi menjadi *frame* GTP sesuai dengan MTU yaitu 1500 byte. Karena untuk *segment* data 323,625 byte memiliki *total frame* MAC sebesar 456,75 byte yang berarti tidak melebihi MTU maka *frame* tersebut langsung dienkapsulasi sesuai dengan persamaan 3.18 yaitu :

$$\begin{aligned} W_{GTP} &= W_{total frame MAC} + header GTP + header UDP + header IP \\ &= 456,75 \text{ byte} + 8 \text{ byte} + 8 \text{ byte} + 40 \text{ byte} \\ &= 512,75 \text{ byte} \end{aligned}$$

Untuk *segment* data 1523,625 byte karena memiliki *total frame* MAC sebesar 1896,813 byte maka akan dilakukan segmentasi sesuai dengan persamaan 3.15 yaitu :

$$N_{frameGTP} = \frac{W_{totalframeMAC}}{MTU} = \frac{1896,813byte}{1500byte} = 1,264 \text{ buah}$$

Dan dengan persamaan 3.16 diperoleh :

$$W_{frame GTP} = MTU_{GTP} + header GTP + header UDP + header IP$$

$$\begin{aligned}
 &= 1500 \text{ byte} + 8 \text{ byte} + 8 \text{ byte} + 40 \text{ byte} \\
 &= 1556 \text{ byte}
 \end{aligned}$$

Sehingga jumlah *total frame* GTP yang siap untuk ditransmisikan lewat AAL5 sesuai dengan persamaan 3.17 adalah :

$$\begin{aligned}
 W_{total \text{ frame } GTP} &= N_{frame \text{ GTP}} \times W_{frame \text{ GTP}} \\
 &= 1,264 \times 1556 \text{ byte} \\
 &= 1967,628 \text{ byte}
 \end{aligned}$$

Pada *layer AAL5 frame* GTP akan diberi *header* sebelum disegmentasi menjadi *fixed-size cells* (53 byte) pada *layer ATM*. Besar paket AAL5 yang siap untuk ditransmisikan pada *layer ATM* sesuai dengan persamaan 3.19 adalah :

$$\begin{aligned}
 W_{AAL5} &= W_{total \text{ frame } GTP} + \text{header PAD} + \text{header CT} \\
 &= 512,75 \text{ byte} + 48 \text{ byte} + 8 \text{ byte} \\
 &= 568,75 \text{ byte}
 \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 2023,628 byte.

Pada *layer ATM* paket AAL5 akan disegmentasi menjadi *fixed-size cells* ATM sebesar 53 byte yang terdiri dari *payload* ATM sebesar 48 byte dan *header* ATM sebesar 8 byte (Stallings). Sesuai dengan persamaan 3.20 diperoleh :

$$N_{frame \text{ ATM}} = \frac{W_{AAL5}}{48 \text{ byte}} = \frac{568,75 \text{ byte}}{48 \text{ byte}} = 11,848 \text{ buah}$$

Sehingga jumlah *total frame* ATM yang siap untuk ditransmisikan ke SGSN sesuai dengan persamaan 3.21 adalah :

$$\begin{aligned}
 W_{total \text{ frame } ATM} &= N_{frame \text{ ATM}} \times \text{fixed-sizes cells } ATM \\
 &= 11,848 \times 53 \text{ byte} \\
 &= 627,994 \text{ byte}
 \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 2234,422 byte.

Nilai *delay* enkapsulasi pada UTRAN untuk *segment* data 323,625 byte saat lintasan *forward* sesuai dengan persamaan 3.22 yaitu :

$$\begin{aligned}
 t_{E2} &= \frac{W_{total \text{ frame } ATM} \times 8}{V_{UTRAN}} \\
 &= \frac{627,994 \times 8}{155 \times 10^6} = \frac{5023,958 \text{ bit}}{155 \times 10^6 \text{ bps}} = 3,241 \times 10^{-5} \text{ s}
 \end{aligned}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi dan dekapsulasi yang terjadi pada UTRAN saat lintasan *forward* untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.3.

Tabel 5.3 Hasil analisis *delay* enkapsulasi dan dekapsulasi pada UTRAN saat lintasan *forward*

No.	Panjang Paket <i>Videophone</i>	<i>Delay</i> Enkapsulasi	<i>Delay</i> Dekapsulasi
1.	323,625 byte	$3,241.10^{-5}$ s	$2.35742.10^{-5}$ s
2.	563,625 byte	$4,882.10^{-5}$ s	$3.84387.10^{-5}$ s
3.	803,625 byte	$6,523.10^{-5}$ s	$5.33032.10^{-5}$ s
4.	1043,625 byte	$8,165.10^{-5}$ s	$6.81677.10^{-5}$ s
5.	1283,625 byte	$9,829.10^{-5}$ s	$8.30323.10^{-5}$ s
6.	1523,625 byte	$1,153.10^{-4}$ s	$9.79001.10^{-5}$ s

Sumber : Perhitungan

➤ **SGSN (*Serving GPRS Support Node*) forward**

Pada SGSN paket yang diterima dari UTRAN akan mengalami proses dekapsulasi. Nilai *delay* dekapsulasi pada SGSN untuk *segment* data 323,625 byte saat lintasan *forward* didapatkan dengan persamaan 3.23 yaitu:

$$t_{D2} = \frac{W_{totalframeATM}}{V_{SGSN}} = \frac{5023,9583 \text{ bit}}{1,25 \times 10^9 \text{ bps}} = 4.01917 \times 10^{-6} \text{ s}$$

Frame yang akan dikirimkan dari SGSN ke GGSN akan mengalami proses enkapsulasi dengan menambahkan *header* GTP, UDP, IP (3GPP TS 29.060 Rel.5) dan CRC pada L2 sebelum ditransmisikan ke GGSN, dengan persamaan 3.24 diperoleh :

$$\begin{aligned} W_{SGSN} &= W_{total \ frame \ ATM} + \text{header GTP} + \text{header UDP} + \text{header IP} + \text{CRC} \\ &= 627,994 \text{ byte} + 8 \text{ byte} + 8 \text{ byte} + 40 \text{ byte} + 2 \text{ byte} \\ &= 685,994 \text{ byte} \end{aligned}$$

Untuk *segment* data 1523,625 byte karena memiliki *total frame* ATM sebesar 2234,422 byte maka akan dilakukan segmentasi sesuai dengan persamaan 3.25 yaitu :

$$N_{frameSGSN} = \frac{W_{totalframeATM}}{MTU_{Ethernet}} = \frac{2234,422byte}{1500byte} = 1,489 \text{ buah}$$

Selanjutnya masing-masing *frame* tersebut dienkapsulasi sesuai dengan persamaan 3.26 yaitu :

$$\begin{aligned} W_{frameSGSN} &= MTU_{Ethernet} + \text{header GTP} + \text{header UDP} + \text{header IP} + \text{CRC} \\ &= 1500 \text{ byte} + 8 \text{ byte} + 8 \text{ byte} + 40 \text{ byte} + 2 \text{ byte} \\ &= 1558 \text{ byte} \end{aligned}$$

Sehingga jumlah *total frame* SGSN untuk *segment* data 1523,625 byte yang siap untuk ditransmisikan ke GGSN adalah :

$$\begin{aligned} W_{total frame SGSN} &= N_{frame SGSN} \times W_{frameSGSN} \\ &= 1,489 \times 1558 \text{ byte} \\ &= 2320,820 \text{ byte} \end{aligned}$$

Nilai *delay* enkapsulasi pada SGSN untuk *segment* data 323,625 byte saat lintasan *forward* yang terjadi didapatkan dengan persamaan 3.28 yaitu:

$$\begin{aligned} t_{E3} &= \frac{W_{totalframeSGSN} \times 8}{V_{SGSN}} \\ &= \frac{685,994 \times 8}{1,25 \times 10^9} = \frac{5487,958bit}{1,25 \times 10^9 bps} = 4,39 \times 10^{-6} s \end{aligned}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi dan dekapsulasi yang terjadi pada SGSN saat lintasan *forward* untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.4.

Tabel 5.4 Hasil analisis *delay* enkapsulasi dan dekapsulasi pada SGSN saat lintasan *forward*

No.	Panjang Paket Videophone	Delay Enkapsulasi	Delay Dekapsulasi
1.	323,625 byte	4,39037.10 ⁻⁶ s	4,01917.10 ⁻⁶ s
2.	563,625 byte	6,42557.10 ⁻⁶ s	6,05437.10 ⁻⁶ s
3.	803,625 byte	8,46077.10 ⁻⁶ s	8,08957.10 ⁻⁶ s
4.	1043,625 byte	1,05163.10 ⁻⁵ s	1,01248.10 ⁻⁵ s
5.	1283,625 byte	1,266.10 ⁻⁵ s	1,21887.10 ⁻⁵ s
6.	1523,625 byte	1,48533.10 ⁻⁵ s	1,43003.10 ⁻⁵ s

Sumber : Perhitungan



➤ **GGSN (Gateway GPRS Support Node)**

Pada GGSN paket yang diterima dari SGSN akan mengalami proses dekapsulasi. Pada skripsi ini GGSN yang digunakan yaitu dari Cisco dengan type *Cisco 7600 Series Router* dengan teknologi *Gigabit Ethernet* dengan transmisi data sebesar 10 Gbps. Nilai *delay* dekapsulasi pada GGSN untuk *segment* data 323,625 byte saat lintasan *forward* diperoleh dengan persamaan 3.29 yaitu:

$$t_{D3} = \frac{W_{totalframeSGSN}}{V_{GGSN}} = \frac{5487,9583bit}{10 \times 10^9 bps} = 5,4879 \times 10^{-7} s$$

Setelah *frame* melalui semua *node* pada jaringan UMTS untuk lintasan *forward*, selanjutnya dilakukan analisis dan perhitungan untuk lintasan *reverse* yaitu dari GGSN (*Gateway GPRS Support Node*) menuju MT (*Mobile Terminal*) penerima.

Frame yang akan dikirimkan dari GGSN ke SGSN akan mengalami proses enkapsulasi dengan menggunakan 3.30 yaitu :

$$\begin{aligned} W_{GGSN} &= W_{totalframeSGSN} + header\ GTP + header\ UDP + header\ IP + CRC \\ &= 685,994\ byte + 8\ byte + 40\ byte + 40\ byte + 2\ byte \\ &= 743,994\ byte \end{aligned}$$

Untuk *segment* data 1523,625 byte karena memiliki *total frame* SGSN sebesar 2320,830 byte maka akan dilakukan segmentasi dengan persamaan 3.31 yaitu:

$$N_{frameGGSN} = \frac{W_{totalframeSGSN}}{MTU_{Ethernet}} = \frac{2320,830byte}{1500byte} = 1,547\ buah$$

Dengan menggunakan persamaan 3.32 *frame* tersebut dienkapsulasi sehingga diperoleh:

$$\begin{aligned} W_{frameGGSN} &= 1500\ byte + 8\ byte + 40\ byte + 40\ byte + 2\ byte \\ &= 1558\ byte \end{aligned}$$

Sehingga jumlah *total frame* GGSN yang siap untuk ditansmisikan ke SGSN adalah :

$$\begin{aligned} W_{totalframeGGSN} &= N_{frameGGSN} \times W_{frameGGSN} \\ &= 1,547 \times 1558 \\ &= 2410,558\ byte \end{aligned}$$

Dengan menggunakan persamaan 3.34 maka diperoleh nilai *delay* enkapsulasi pada GGSN untuk *segment* data 323,625 byte saat lintasan *reverse* yaitu :

$$t_{E4} = \frac{W_{totalframeGGSN} \times 8}{V_{GGSN}} = \frac{743,994 \times 8}{10 \times 10^{10}} = \frac{5951,958 \text{ bit}}{10 \times 10^{10} \text{ bps}} = 5,951 \times 10^{-7} \text{ s}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi saat lintasan *reverse* dan dekapsulasi saat lintasan *forward* yang terjadi pada GGSN untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.5.

Tabel 5.5 Hasil analisis *delay* enkapsulasi (*reverse*) dan dekapsulasi (*forward*) pada GGSN

No.	Panjang Paket Videophone	Delay Enkapsulasi	Delay Dekapsulasi
1.	323,625 byte	$5,95196.10^{-7} \text{ s}$	$5,48796.10^{-7} \text{ s}$
2.	563,625 byte	$8,49596.10^{-7} \text{ s}$	$8,03196.10^{-7} \text{ s}$
3.	803,625 byte	$1,104.10^{-6} \text{ s}$	$1,0576.10^{-6} \text{ s}$
4.	1043,625 byte	$1,36536.10^{-6} \text{ s}$	$1,31453.10^{-6} \text{ s}$
5.	1283,625 byte	$1,64368.10^{-6} \text{ s}$	$1,58249.10^{-6} \text{ s}$
6.	1523,625 byte	$1,92845.10^{-6} \text{ s}$	$1,85666.10^{-6} \text{ s}$

Sumber : Perhitungan

➤ **SGSN (Serving GPRS Support Node) reverse**

Nilai *delay* dekapsulasi pada SGSN untuk *segment* data 323,625 byte saat lintasan *reverse* didapatkan dengan persamaan 3.35 :

$$t_{D4} = \frac{W_{totalframeGGSN}}{V_{SGSN}} = \frac{5951,958 \text{ bit}}{1,25 \times 10^9 \text{ bps}} = 4,761 \times 10^{-6} \text{ s}$$

Pada SGSN *frame* GGSN dienkapsulasi sesuai dengan persamaan 3.37 :

$$\begin{aligned} W_{GTP} &= W_{total \text{ frame } GGSN} + \text{header GTP} + \text{header UDP} + \text{header IP} \\ &= 743,994 \text{ byte} + 8 \text{ byte} + 8 \text{ byte} + 40 \text{ byte} \\ &= 799,994 \text{ byte} \end{aligned}$$

Untuk *segment* data 1523,625 byte karena memiliki *total frame* GGSN sebesar 2410,558 byte maka akan dilakukan segmentasi sesuai dengan persamaan 3.36 yaitu :

$$N_{frameGTP} = \frac{W_{totalframeGGSN}}{MTU} = \frac{2410,558 \text{ byte}}{1500 \text{ byte}} = 1,607 \text{ buah}$$



Dengan persamaan 3.16 diperoleh :

$$\begin{aligned}W_{frame\ GTP} &= MTU_{GTP} + header\ GTP + header\ UDP + header\ IP \\ &= 1500\ byte + 8\ byte + 8\ byte + 40\ byte \\ &= 1556\ byte\end{aligned}$$

Sehingga jumlah *total frame* GTP yang siap untuk ditransmisikan lewat AAL5 sesuai dengan persamaan 3.17 adalah :

$$\begin{aligned}W_{total\ frame\ GTP} &= N_{frame\ GTP} \times W_{frame\ GTP} \\ &= 1,607 \times 1556\ byte \\ &= 2500,552\ byte\end{aligned}$$

Pada *layer AAL5 frame* GTP akan diberi *header* sebelum disegmentasi menjadi *fixed-size cells* (53 byte) pada *layer ATM*. Besar paket AAL5 yang siap untuk ditransmisikan pada *layer ATM* sesuai dengan persamaan 3.19 adalah :

$$\begin{aligned}W_{AAL5} &= W_{total\ frame\ GTP} + header\ PAD + header\ CT \\ &= 799,994\ byte + 48\ byte + 8\ byte \\ &= 855,994\ byte\end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 2556,552 byte.

Pada *layer ATM* paket AAL5 akan disegmentasi menjadi *fixed-size cells* ATM sebesar 53 byte yang terdiri dari *payload* ATM sebesar 48 byte dan *header* ATM sebesar 8 byte (Stallings). Sesuai dengan persamaan 3.20 diperoleh :

$$N_{frame\ ATM} = \frac{W_{AAL5}}{48\ byte} = \frac{855,994\ byte}{48\ byte} = 17,833\ \text{buah}$$

Sehingga jumlah *total frame* ATM yang siap untuk ditransmisikan ke UTRAN sesuai dengan persamaan 3.21 adalah :

$$\begin{aligned}W_{total\ frame\ ATM} &= N_{frame\ ATM} \times fixed\text{-}sizes\ cells\ ATM \\ &= 17,833 \times 53\ byte \\ &= 945,160\ byte\end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 2822,860 byte.

Nilai *delay* enkapsulasi pada SGSN untuk *segment* data 323,625 byte saat lintasan *reverse* didapatkan dengan persamaan 3.38 yaitu :

$$t_{E5} = \frac{W_{totalframeATM} \times 8}{V_{SGSN}}$$

$$= \frac{945,160 \times 8}{1,25 \times 10^9} = \frac{7561,287 \text{ bit}}{1,25 \times 10^9 \text{ bps}} = 6,049 \times 10^{-6} \text{ s}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi dan dekapsulasi saat lintasan *reverse* yang terjadi pada SGSN untuk panjang paket 323,625 byte, 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.6.

Tabel 5.6 Hasil analisis *delay* enkapsulasi dan dekapsulasi pada SGSN saat lintasan *reverse*

No.	Panjang Paket Videophone	Delay Enkapsulasi	Delay Dekapsulasi
1.	323,625 byte	6,04903.10 ⁻⁶ s	4,76157.10 ⁻⁶ s
2.	563,625 byte	8,29623.10 ⁻⁶ s	6,79677.10 ⁻⁶ s
3.	803,625 byte	1,05434.10 ⁻⁵ s	8,83197.10 ⁻⁶ s
4.	1043,625 byte	1,29067.10 ⁻⁵ s	1,09229.10 ⁻⁵ s
5.	1283,625 byte	1,5457.10 ⁻⁵ s	1,31495.10 ⁻⁵ s
6.	1523,625 byte	1,80663.10 ⁻⁵ s	1,54276.10 ⁻⁵ s

Sumber : Perhitungan

➤ **UTRAN (UMTS Terrestrial Radio Access Network) reverse**

Nilai *delay* dekapsulasi pada UTRAN untuk *segment* data 323,625 byte saat lintasan *reverse* didapatkan dengan persamaan 3.39 yaitu :

$$t_{D5} = \frac{W_{totalframeATM}}{V_{UTRAN}} = \frac{7561,287 \text{ bit}}{155 \times 10^6 \text{ bps}} = 4,878 \times 10^{-5} \text{ s}$$

Pada UTRAN *frame* ATM disegmentasi menjadi PDCP Data PDU (*Protocol Data Unit*) sesuai dengan persamaan 3.40 dan format seperti pada gambar 3.8 yaitu dengan menambahkan *header* PDU Type sebesar 3 bits dan PID (*Packet Identifier*) sebesar 5 bits jadi total *header* sebesar 8 bits atau 1 byte (3GPP TS 25.323 Rel.5) dan diperoleh :

$$W_{PDCP} = W_{totalframeATM} + \text{Header}_{PDCP}$$

$$W_{PDCP} = 945,160 \text{ byte} + 1 \text{ byte}$$

$$= 946,160 \text{ byte}$$

Untuk *segment* data 1523,625 byte karena melebihi jumlah MTU (*Maximum Transmit Unit*) sebesar 1500 byte maka paket mengalami segmentasi menjadi beberapa buah *frame* sesuai persamaan 3.41 yaitu :

$$N_{frame-PDCP} = \frac{W_{totalframeATM}}{MTU_{PDCP}} = \frac{2822,860byte}{1500byte} = 1,881 \text{ buah}$$

Dengan menggunakan persamaan 3.6 diperoleh panjang *frame* PDCP sebesar:

$$\begin{aligned} W_{frame PDCP} &= MTU_{PDCP} + Header_{PDCP} \\ &= 1500 \text{ byte} + 1 \text{ byte} \\ &= 1501 \text{ byte} \end{aligned}$$

Jumlah total *frame* PDCP Data PDU yang dikirimkan ke RLC (*Radio Link Control*) sesuai dengan persamaan 3.7 yaitu:

$$\begin{aligned} W_{frame PDCP total} &= N_{frame PDCP} \times W_{frame PDCP} \\ &= 1,881 \times 1501 \text{ byte} \\ &= 2824,742 \text{ byte} \end{aligned}$$

Pada *layer* RLC, PDCP Data PDU disegmentasi menjadi RLC PDU *fixed size* sebesar 40 byte, dengan persamaan 3.8 diperoleh :

$$N_{frame-RLC} = \frac{W_{PDCP}}{40byte} = \frac{946,160byte}{40byte} = 23,654 \text{ buah}$$

Kemudian dengan persamaan 3.9 diperoleh panjang *frame* RLC sebesar :

$$\begin{aligned} W_{frame RLC} &= 40 \text{ byte} + header RLC \\ &= 40 \text{ byte} + 2 \text{ byte} \\ &= 42 \text{ byte} \end{aligned}$$

Sehingga jumlah total *frame* RLC yang dikirimkan ke MAC adalah :

$$\begin{aligned} W_{total frame RLC} &= N_{frame RLC} \times W_{frame RLC} \\ &= 23,654 \times 42 \text{ byte} \\ &= 993,468 \text{ byte} \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 2965,979 byte

Pada *layer* MAC, untuk menjadi paket data yang siap untuk ditransmisikan lewat *physical layer* maka panjang *frame* RLC PDU akan ditambah *header* MAC sebesar 3 byte dan CRC 2 byte sesuai dengan persamaan 3.11 diperoleh :

$$\begin{aligned}
 W_{frame\ MAC} &= \text{Header MAC} + W_{frame\ RLC} + \text{CRC} \\
 &= 3\ \text{byte} + 42\ \text{byte} + 3\ \text{byte} \\
 &= 48\ \text{byte}
 \end{aligned}$$

Sehingga jumlah total *frame* MAC PDU yang siap untuk ditransmisikan lewat *physical layer* dengan persamaan 3.12 adalah :

$$\begin{aligned}
 W_{total\ frame\ MAC} &= N_{frame\ RLC} \times W_{frame\ MAC} \\
 &= 23,654 \times 48\ \text{byte} \\
 &= 1135,393\ \text{byte}
 \end{aligned}$$

Dengan cara yang sama untuk *segment* data 1523,625 byte diperoleh sebesar 3389,690 byte.

Nilai *delay* enkapsulasi pada UTRAN untuk *segment* data 323,625 byte saat lintasan *reverse* didapatkan dengan persamaan 3.42 yaitu :

$$\begin{aligned}
 t_{E6} &= \frac{W_{total\ frame\ MAC} \times 8}{V_{UTRAN}} \\
 &= \frac{1135,393 \times 8}{155 \times 10^6} = \frac{9083,144\ \text{bit}}{155 \times 10^6\ \text{bps}} = 5,86 \times 10^{-5}\ \text{s}
 \end{aligned}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi dan dekapsulasi saat lintasan *reverse* yang terjadi pada UTRAN untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.7.

Tabel 5.7 Hasil analisis *delay* enkapsulasi dan dekapsulasi pada UTRAN saat lintasan *reverse*

No.	Panjang Paket Videophone	Delay Enkapsulasi	Delay Dekapsulasi
1.	323,625 byte	$5,86009 \cdot 10^{-5}\ \text{s}$	$4,87825 \cdot 10^{-5}\ \text{s}$
2.	563,625 byte	$8,0348 \cdot 10^{-5}\ \text{s}$	$6,69051 \text{E} \cdot 10^{-5}\ \text{s}$
3.	803,625 byte	$1,02101 \cdot 10^{-4}\ \text{s}$	$8,50277 \cdot 10^{-5}\ \text{s}$
4.	1043,625 byte	$1,24987 \cdot 10^{-4}\ \text{s}$	$1,04086 \cdot 10^{-4}\ \text{s}$
5.	1283,625 byte	$1,49684 \cdot 10^{-4}\ \text{s}$	$1,24653 \cdot 10^{-4}\ \text{s}$
6.	1523,625 byte	$1,74952 \cdot 10^{-4}\ \text{s}$	$1,45696 \cdot 10^{-4}\ \text{s}$

Sumber : Perhitungan



➤ **Mobile Terminal reverse**

Nilai *delay* dekapsulasi pada MT (*Mobile Terminal*) untuk *segment* data 323,625 byte saat lintasan *reverse* dapat dihitung dengan persamaan 3.43 yaitu :

$$t_{D6} = \frac{W_{totalframeMAC} \times 8}{V_{MT}} = \frac{1135,393 \times 8}{384000} = 2,36 \times 10^{-2} s$$

Dengan cara yang sama, hasil analisis *delay* dekapsulasi saat lintasan *reverse* yang terjadi pada MT untuk 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.8.

Tabel 5.8 Hasil analisis *delay* dekapsulasi pada *Mobile Terminal* (MT) saat lintasan *reverse*

No.	Panjang Paket Videophone	Delay Dekapsulasi
1.	323,625 byte	$2,3654023 \cdot 10^{-2} s$
2.	563,625 byte	$3,2432148 \cdot 10^{-2} s$
3.	803,625 byte	$4,121273 \cdot 10^{-2} s$
4.	1043,625 byte	$5,0450353 \cdot 10^{-2} s$
5.	1283,625 byte	$6,0419121 \cdot 10^{-2} s$
6.	1523,625 byte	$7,0618561 \cdot 10^{-2} s$

Sumber : Perhitungan

Dari hasil analisis diatas maka total *delay* enkapsulasi dan dekapsulasi dari MT pengirim ke MT penerima untuk *segment* data 323,625 byte adalah :

$$\begin{aligned} t_{E \text{ total}} &= t_{E1} + t_{E2} + t_{E3} + t_{E4} + t_{E5} + t_{E6} \\ &= (9,515 \cdot 10^{-3} + 3,241 \cdot 10^{-5} + 4,390 \cdot 10^{-6} + 5,951 \cdot 10^{-7} + 6,049 \cdot 10^{-6} + \\ &\quad 5,860 \cdot 10^{-5}) s \\ &= 9,617 \cdot 10^{-3} s \end{aligned}$$

$$\begin{aligned} t_{D \text{ total}} &= t_{D1} + t_{D2} + t_{D3} + t_{D4} + t_{D5} + t_{D6} \\ &= (2,357 \cdot 10^{-5} + 4,019 \cdot 10^{-6} + 5,487 \cdot 10^{-7} + 4,761 \cdot 10^{-6} + 4,878 \cdot 10^{-5} + \\ &\quad 2,365 \cdot 10^{-2}) s \\ &= 2,373 \cdot 10^{-2} s \end{aligned}$$

$$\begin{aligned} t_{\text{proc}} &= t_{E \text{ total}} + t_{D \text{ total}} \\ &= 9,617 \cdot 10^{-3} s + 2,373 \cdot 10^{-2} s \\ &= 3,335 \cdot 10^{-2} s \end{aligned}$$

Dengan cara yang sama, hasil analisis *delay* enkapsulasi dan dekapsulasi dari MT pengirim ke MT penerima untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.9.

Tabel 5.9 Hasil analisis *delay* enkapsulasi dan dekapsulasi total

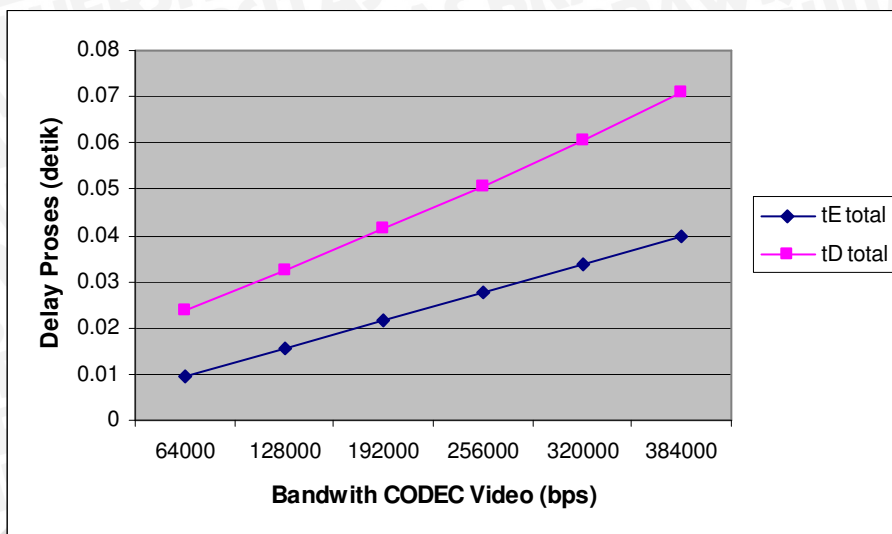
No.	Panjang Paket Videophone	Delay Enkapsulasi Total	Delay Dekapsulasi Total	Delay Proses
1.	323,625 byte	0,009617673 s	0,023735709 s	0,033353382 s
2.	563,625 byte	0,01566037 s	0,032551146 s	0,048211516 s
3.	803,625 byte	0,021703073 s	0,04136904 s	0,063072113 s
4.	1043,625 byte	0,027747051s	0,050644969 s	0,07839202 s
5.	1283,625 byte	0,033793365 s	0,060653727 s	0,094447092 s
6.	1523,625 byte	0,039842077 s	0,070893741 s	0,110735818 s

Sumber : Perhitungan

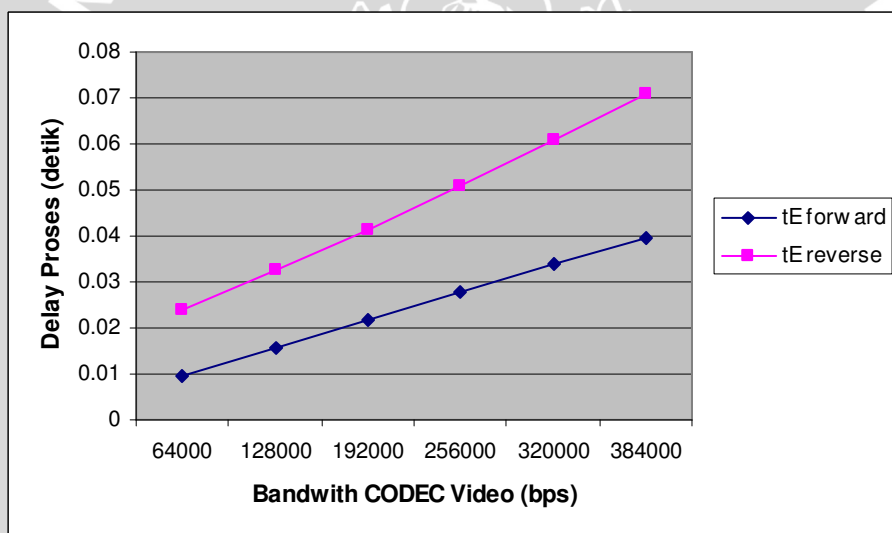
Tabel 5.10 Hasil analisis *delay forward* dan *reverse* total

No.	Panjang Paket Videophone	Delay Forward Total	Delay Reverse Total
1.	323,625 byte	0,00958057 s	0,023772812 s
2.	563,625 byte	0,015616172 s	0,032595344 s
3.	803,625 byte	0,021651775 s	0,041420338 s
4.	1043,625 byte	0,0276874 s	0,050704621 s
5.	1283,625 byte	0,033723384 s	0,060723708 s
6.	1523,625 byte	0,039761187 s	0,070974631 s

Sumber : Perhitungan



Gambar 5.5 Grafik hubungan *bandwidth CODEC video* dengan besarnya *delay proses* pada Jaringan UMTS



Gambar 5.6 Grafik hubungan *bandwidth CODEC video* dengan besarnya *delay proses* pada jaringan UMTS

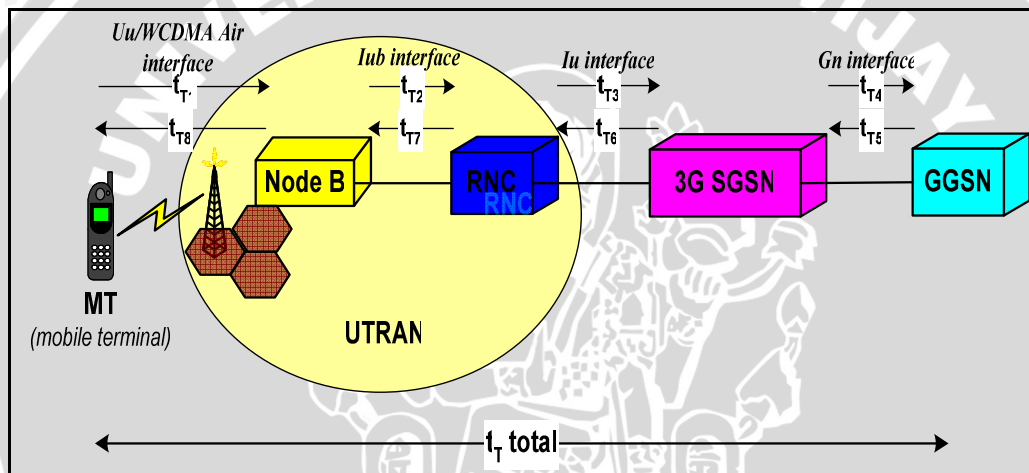
Dari hasil analisis *delay proses* di atas maka dapat disimpulkan :

- Semakin besar *bandwidth CODEC video* yang digunakan untuk aplikasi *videophone*, maka *delay* enkapsulasi dan dekapsulasi juga semakin besar. Hal ini dikarenakan semakin besar *bandwidth CODEC video* yang digunakan semakin besar pula *segment* datanya.
- *Delay* enkapsulasi dan dekapsulasi terbesar dicapai saat pengiriman *segment* data 1503,625 byte untuk *bandwidth CODEC video* 384 kbps yaitu sebesar

0,0398 s dan 0,0708 s. Sedangkan *delay* enkapsulasi dan dekapsulasi terkecil dicapai saat pengiriman *segment* data 303,625 byte untuk *bandwidth CODEC video* 64 kbps yaitu sebesar 0,00961 s dan 0,0237 s.

5.2.3.2 Delay transmisi

Dalam analisis ini dilakukan penghitungan *delay* transmisi *end-to-end* pada hubungan antara MT-Node B (*Uu/WCDMA Air interface*), Node B-RNC (*Iub interface*), RNC-SGSN (*Iu interface*) dan SGSN-GGSN (*Gn Interface*) seperti yang digambarkan pada gambar 5.7 yang dihitung dalam dua lintasan yaitu *forward* dan *reverse*. Berikut perhitungan masing-masing *delay* transmisi untuk tiap *node* dan tiap lintasan.



Gambar 5.7 Analisis *delay* transmisi pada jaringan UMTS
sumber : Analisis

(a) Uu Interface lintasan forward

Antara MT dengan Node B dihubungkan dengan menggunakan *Uu interface* dengan kecepatan sesuai dengan *data rate*. *Delay* transmisi yang terjadi pada *Uu interface* jika menggunakan *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps berdasarkan persamaan 3.47 diperoleh :

$$t_{T1} = \frac{W_{totalframeMAC}}{n \times V_{MT}} = \frac{3654bit}{15 \times 384000bps} = 6,34 \times 10^{-4} s$$

(b) Iub Interface saat lintasan forward

Iub interface menghubungkan antara Node B dan RNC. *Physical layer* untuk *Iub interface* pada jaringan UMTS berbasis IP menggunakan *interface*

sistem transmisi digital STM-1 yang memberikan kecepatan data sebesar 155 Mbps [3GPP TS 25.411]. Sehingga nilai *delay* transmisinya didapatkan dengan persamaan 3.48 yaitu :

$$t_{T2} = \frac{W_{total\ frame\ ATM}}{V_{Iub\ interface}} = \frac{5023,958bit}{155 \times 10^6\ bps} = 3,241 \times 10^{-5}\ s$$

(c) Iu Interface saat lintasan forward

Antara RNC dengan SGSN dihubungkan dengan menggunakan *interface Gigabit Ethernet* yang memberikan kecepatan data sebesar 1 Gbps. Sehingga nilai *delay* transmisi yang terjadi pada *Iu interface* diperoleh dengan persamaan 3.49 yaitu :

$$t_{T3} = \frac{W_{total\ frame\ ATM}}{V_{Iu\ interface}} = \frac{5023,958bit}{10^9\ bps} = 5,023 \times 10^{-6}\ s$$

(d) Gn interface saat lintasan forward

Antara SGSN dengan GGSN dihubungkan dengan menggunakan *ethernet* sebagai *interface* yang memiliki kecepatan transmisi data sebesar 1 Gbps. Sehingga *delay* transmisi yang terjadi pada *Gn interface* adalah sesuai dengan persamaan 3.50 yaitu :

$$t_{T4} = \frac{W_{total\ frame\ SGSN}}{V_{ethernet}} = \frac{5487,958bit}{10^9} = 5,487 \times 10^{-6}\ s$$

(e) Gn Interface saat lintasan reverse

Antara GGSN dengan SGSN dihubungkan dengan menggunakan *ethernet* sebagai *interface* yang memiliki kecepatan transmisi data sebesar 1 Gbps. Sehingga *delay* transmisi yang terjadi pada *Gn interface* pada saat lintasan *reverse* ditentukan dengan persamaan 3.51 yaitu :

$$t_{T5} = \frac{W_{total\ frame\ GGSN}}{V_{ethernet}} = \frac{5951,958bit}{10^9} = 5,951 \times 10^{-6}\ s$$

(f) Iu Interface saat lintasan reverse

Antara SGSN dengan RNC dihubungkan dengan menggunakan *interface sistem Gigabit ethernet* yang memberikan kecepatan data sebesar 1Gbps. Sehingga *delay* transmisi yang terjadi pada *Iu interface* adalah didapatkan dengan persamaan 3.52 yaitu :

$$t_{T6} = \frac{W_{totalframeATM}}{V_{lubinterface}} = \frac{7561,287bit}{10^9 bps} = 7,561 \times 10^{-6} s$$

(g) Iub Interface saat lintasan reverse

Iub interface menghubungkan antara RNC dengan Node B. *Physical layer* untuk *Iub interface* pada jaringan UMTS berbasis IP menggunakan *interface* sistem transmisi digital STM-1 yang memberikan kecepatan data sebesar 155 Mbps [3GPP TS 25.411]. Sehingga nilai *delay* transmisinya yang didapatkan dengan persamaan 3.53 yaitu:

$$t_{T7} = \frac{W_{totalframeATM}}{V_{lubinterface}} = \frac{7561,287bit}{155 \times 10^6 bps} = 4,878 \times 10^{-5} s$$

(h) Uu Interface lintasan reverse

Antara Node B dengan MT dihubungkan dengan menggunakan *Uu interface* dengan kecepatan sesuai *data rate*. Sehingga *delay* transmisi yang terjadi pada *Uu interface* dengan persamaan 3.54 yaitu :

$$t_{T8} = \frac{W_{totalframeMAC}}{n \times V_{MT}} = \frac{9083,144bit}{15 \times 384000bps} = 1,576 \times 10^{-3} s$$

(i) Delay transmisi total

Nilai *delay* transmisi untuk lintasan *forward* ditentukan dengan persamaan 3.55 yaitu :

$$\begin{aligned} t_{Tforward} &= t_{T1} + t_{T2} + t_{T3} + t_{T4} \\ &= (6,343 \cdot 10^{-4} + 3,241 \cdot 10^{-5} + 5,023 \cdot 10^{-6} + 5,487 \cdot 10^{-6}) s \\ &= 6,773 \times 10^{-4} s \end{aligned}$$

Nilai *delay* transmisi untuk lintasan *reverse* ditentukan dengan persamaan 3.56 yaitu:

$$\begin{aligned} t_{Treverse} &= t_{T5} + t_{T6} + t_{T7} + t_{T8} \\ &= (5,951 \cdot 10^{-6} + 7,561 \cdot 10^{-6} + 4,878 \cdot 10^{-5} + 1,576 \cdot 10^{-3}) \\ &= 1,639 \times 10^{-3} s \end{aligned}$$

Nilai *delay* transmisi total ditentukan dengan persamaan 3.57 yaitu :

$$\begin{aligned} t_{TTotal} &= t_{Tforward} + t_{Treverse} \\ &= 6,773 \cdot 10^{-4} s + 1,639 \cdot 10^{-3} s \\ &= 2,316 \times 10^{-3} s \end{aligned}$$

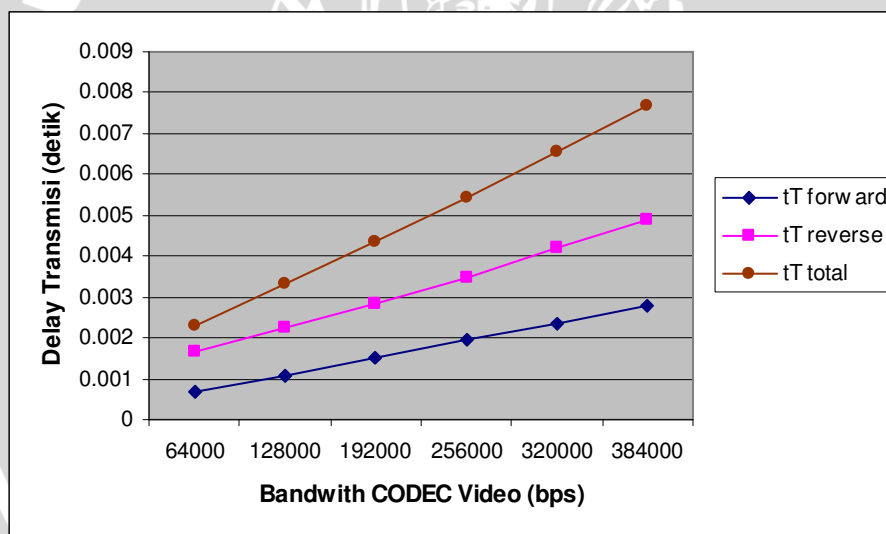


Dengan cara yang sama, hasil analisis *delay* transmisi *end-to-end* dari MT pengirim ke MT penerima untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.11.

Tabel 5.11 Hasil analisis *delay* transmisi total

No.	Panjang Paket Videophone	Delay Transmisi Forward	Delay Transmisi Reverse	Delay Transmisi Total
1.	323,625 byte	0,0006773 s	0,0016392 s	0,0023165 s
2.	563,625 byte	0,0010988 s	0,0022479 s	0,0033467 s
3.	803,625 byte	0,0015203 s	0,0028567 s	0,0043770 s
4.	1043,625 byte	0,0019418 s	0,0034972 s	0,0054390 s
5.	1283,625 byte	0,0023637 s	0,0041883 s	0,0065520 s
6.	1523,625 byte	0,0027862 s	0,0048954 s	0,0076816 s

Sumber : Perhitungan



Gambar 5.8 Grafik hubungan *bandwidth CODEC video* dengan besarnya *delay* transmisi pada jaringan UMTS

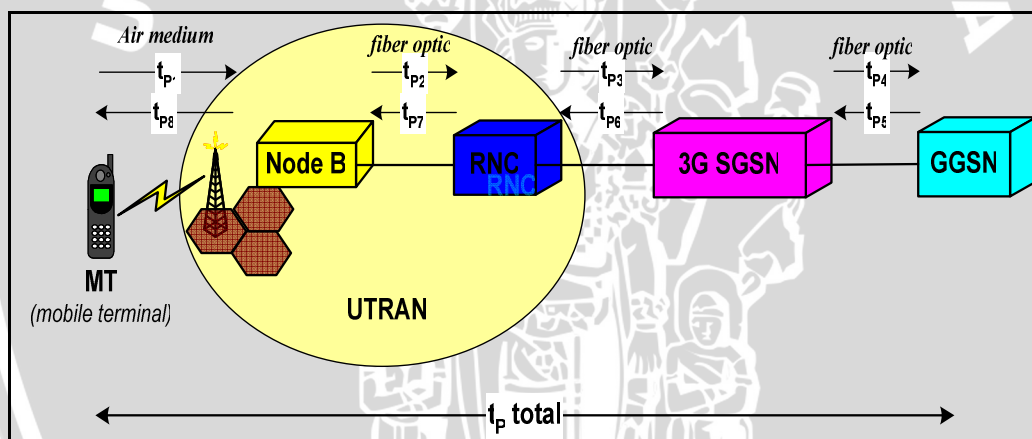
Dari hasil analisis *delay* transmisi di atas maka dapat disimpulkan :

- Semakin besar *bandwidth CODEC video* yang digunakan untuk aplikasi *videophone*, maka *delay* transmisi juga semakin besar. Hal ini dikarenakan semakin besar *bandwidth CODEC video* yang digunakan semakin besar pula *segment* datanya.

- *Delay* transmisi total terbesar dicapai saat pengiriman *segment* data 1503,625 byte untuk *bandwidth CODEC video* 384 kbps yaitu sebesar 0,0076816 s. Sedangkan *delay* transmisi total terkecil dicapai saat pengiriman *segment* data 303,625 byte untuk *bandwidth CODEC video* 64 kbps yaitu sebesar 0,0023165 s.

5.2.3.3 Delay Propagasi

Nilai *delay* propagasi pada analisis ini dihitung pada tiap *node* dengan lintasan *end to end* (MT, Node B, RNC, SGSN, GGSN, SGSN, RNC, Node B, MT) dimana antara MT dengan Node B menggunakan media udara dan antara Node B sampai dengan GGSN menggunakan media *fiber optic* seperti yang digambarkan pada gambar 5.9. Berikut perhitungan masing-masing *delay* transmisi untuk tiap *node* yang dihitung pada lintasan *forward* dan *reverse*.



Gambar 5.9 Analisis *delay* propagasi pada jaringan UMTS
sumber : Analisis

(a) *Delay* propagasi dari MT ke Node B lintasan *forward*

Nilai *delay* propagasi dari MT ke Node untuk lintasan *forward* dihitung dengan menggunakan persamaan 3.60. Pada skripsi ini nilai *radius cell* yang digunakan adalah untuk type sel *micro* dengan daerah operasi urban dan pusat kota sesuai dengan tabel 3.2 yaitu sebesar 0,4 km dengan kecepatan data maksimal 384 Kbps. Sehingga besarnya *delay* propagasi jika menggunakan *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps akan diperoleh sebesar :

$$\begin{aligned}
 t_{p1} &= \frac{d}{c} = \frac{N_{frameMAC} \times R}{c} \\
 &= \frac{9,515 \times 400m}{3 \times 10^8 \text{ m/s}} = 1,268 \times 10^{-5} \text{ s}
 \end{aligned}$$

(b) Delay Propagasi dari Node B ke RNC lintasan forward

Pada Node B dan RNC dihubungkan dengan STM1 dengan media transmisi adalah *fiber optic* dan diasumsikan jarak antara Node B ke RNC adalah 100 m. Sehingga dengan persamaan 3.61 nilai *delay* propagasinya adalah:

$$\begin{aligned}
 t_{p2} &= \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \\
 &= \frac{11,848 \times 100m}{2 \times 10^8 \text{ m/s}} = 5,924 \times 10^{-6} \text{ s}
 \end{aligned}$$

(c) Delay Propagasi dari RNC ke SGSN lintasan forward

Pada RNC ke SGSN dihubungkan dengan media transmisi *fiber optic* dengan jarak antara RNC ke SGSN diasumsikan adalah 25 m. Nilai *delay* propagasi ditentukan dengan persamaan 3.62 yaitu :

$$\begin{aligned}
 t_{p3} &= \frac{d}{c} = \frac{N_{frameATM} \times d}{c} \\
 &= \frac{11,848 \times 25m}{2 \times 10^8 \text{ m/s}} = 1,481 \times 10^{-6} \text{ s}
 \end{aligned}$$

(d) Delay Propagasi dari SGSN ke GGSN lintasan forward

Pada SGSN ke GGSN dihubungkan dengan media transmisi *fiber optic* dengan jarak antara SGSN ke GGSN diasumsikan adalah 25 m. Nilai *delay* propagasi antara SGSN ke GGSN didapatkan dengan menggunakan persamaan 3.63 yaitu :

$$\begin{aligned}
 t_{p4} &= \frac{d}{c} = \frac{N_{frameSGSN} \times d}{c} \\
 &= \frac{1 \times 25m}{2 \times 10^8 \text{ m/s}} = 1,25 \times 10^{-7} \text{ s}
 \end{aligned}$$

(e) Delay Propagasi dari GGSN ke SGSN lintasan reverse

Pada saat lintasan *reverse* dari GGSN ke SGSN sama dengan pada saat lintasan *forward* yaitu panjang 25 m dengan media transmisi *fiber optic*. Sehingga *delay* propagasi sesuai dengan persamaan 3.64 yaitu :

$$t_{p5} = \frac{d}{c} = \frac{N_{frameSGSN} \times d}{c}$$

$$= \frac{1 \times 25m}{2 \times 10^8 m/s} = 1,25 \times 10^{-7} s$$

(f) Delay Propagasi dari SGSN ke RNC saat lintasan reverse

Pada saat lintasan *reverse* dari SGSN ke RNC sama dengan pada saat lintasan *forward* yaitu panjang 25 m dengan media transmisi *fiber optic*. Sehingga *delay* propagasi sesuai dengan persamaan 3.65 yaitu :

$$t_{p6} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c}$$

$$= \frac{17,833 \times 25m}{2 \times 10^8 m/s} = 2,229 \times 10^{-6} s$$

(g) Delay Propagasi dari RNC ke Node B saat lintasan reverse

Pada saat lintasan *reverse* dari RNC ke Node B sama dengan pada saat lintasan *forward* yaitu panjang 100 m dengan media transmisi *fiber optic*. Untuk itu nilai *delay* propagasi ditentukan dengan persamaan 3.66 yaitu :

$$t_{p7} = \frac{d}{c} = \frac{N_{frameATM} \times d}{c}$$

$$= \frac{17,833 \times 100m}{2 \times 10^8 m/s} = 8,916 \times 10^{-6} s$$

(h) Delay Propagasi Node B ke MT lintasan reverse

Pada saat lintasan *forward* dari Node B ke MT menggunakan WCDMA *Air interface*. Untuk *delay* propagasi saat lintasan *reverse* sama dengan saat lintasan *forward* dengan radius R sebesar 0,4 km. Nilai *delay* propagasi ini ditentukan dengan persamaan 3.67 yaitu :

$$t_{p8} = \frac{d}{c} = \frac{N_{frameMAC} \times R}{c}$$

$$= \frac{23,654 \times 400m}{3 \times 10^8 m/s} = 3,153 \times 10^{-5} s$$

(i) Delay Propagasi total

Dari analisis diatas maka akan didapat nilai untuk *delay* propagasi untuk lintasan *forward*, lintasan *reverse*, dan *delay* total.

Delay propagasi untuk lintasan *forward* ditentukan dengan persamaan 3.68 yaitu :

$$\begin{aligned}
 t_{p \text{ forward}} &= t_{p1} + t_{p2} + t_{p3} + t_{p4} \\
 &= (1,268.10^{-5} + 5,924.10^{-6} + 1,481.10^{-6} + 1,25.10^{-7}) \text{ s} \\
 &= 2,021.10^{-5} \text{ s}
 \end{aligned}$$

Delay propagasi untuk lintasan *reverse* ditentukan dengan persamaan 3.69 yaitu :

$$\begin{aligned}
 t_{p \text{ reverse}} &= t_{p5} + t_{p6} + t_{p7} + t_{p8} \\
 &= (1,25.10^{-7} + 2,229.10^{-6} + 8,916.10^{-6} + 3,153.10^{-5}) \text{ s} \\
 &= 4,280.10^{-5} \text{ s}
 \end{aligned}$$

Delay propagasi total ditentukan dengan persamaan 3.70 yaitu :

$$\begin{aligned}
 t_{p \text{ Total}} &= t_{p \text{ forward}} + t_{p \text{ reverse}} \\
 &= 2,021.10^{-5} \text{ s} + 4,280.10^{-5} \text{ s} \\
 &= 6,301.10^{-5}
 \end{aligned}$$

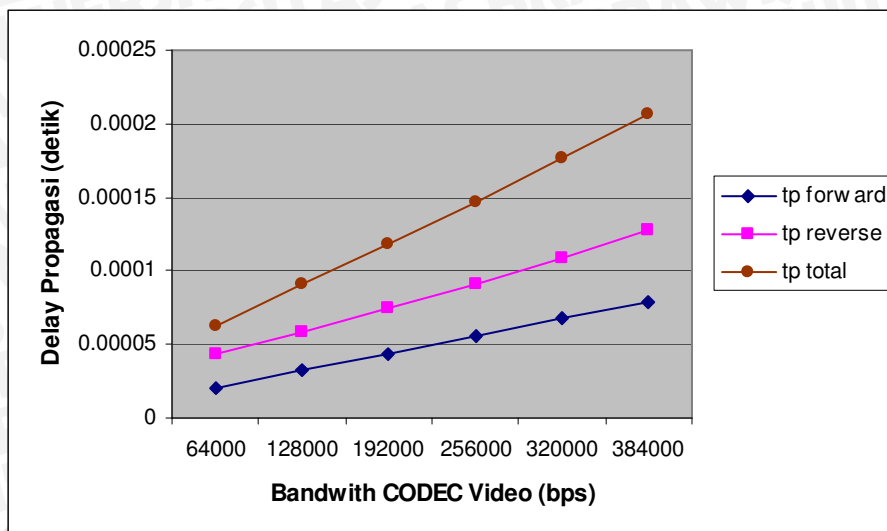
Dengan cara yang sama, hasil analisis *delay propagasi end-to-end* dari MT pengirim ke MT penerima untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte ditunjukkan dalam tabel 5.12.

Tabel 5.12 Hasil analisis *delay propagasi total*

No.	Panjang Paket Videophone	Delay Propagasi Forward	Delay Propagasi Reverse	Delay Propagasi Total
1.	323,625 byte	0,0000202181 s	0,0000428095 s	0,000063027 s
2.	563,625 byte	0,0000319681 s	0,0000586543 s	0,000090622 s
3.	803,625 byte	0,0000437181 s	0,0000745023 s	0,00011822 s
4.	1043,625 byte	0,0000554749 s	0,0000911855 s	0,00014666 s
5.	1283,625 byte	0,0000673047 s	0,000109204 s	0,000176509 s
6.	1523,625 byte	0,0000792248 s	0,00012764 s	0,000206865 s

Sumber : Perhitungan





Gambar 5.10 Grafik hubungan *bandwith CODEC video* dengan besarnya *delay propagasi* pada jaringan UMTS

Dari hasil analisis *delay propagasi* di atas maka dapat disimpulkan :

- Semakin besar *bandwidth CODEC video* yang digunakan untuk aplikasi *videophone*, maka *delay propagasi* juga semakin besar. Hal ini dikarenakan semakin besar *bandwidth CODEC video* yang digunakan semakin besar pula *segment* datanya.
- *Delay propagasi total* terbesar dicapai saat pengiriman *segment* data 1503,625 byte untuk *bandwidth CODEC video* 384 kbps yaitu sebesar $2,068 \times 10^{-4}$ s. Sedangkan *delay propagasi total* terkecil dicapai saat pengiriman *segment* data 303,625 byte untuk *bandwidth CODEC video* 64 kbps yaitu sebesar $7,922 \times 10^{-5}$ s.

5.2.3.4 Delay antrian

Delay antrian yang dianalisis adalah *delay* yang terjadi pada GGSN dengan menggunakan model antrian M/M/1 sebagaimana telah dijelaskan pada bab 3. Dengan menggunakan kecepatan standar GGSN dari Cisco dengan type Cisco 7600 Series Router dengan teknologi *Gigabit Ethernet* dengan transmisi data sebesar 10 Gbps dan panjang data sebesar 1500 byte, maka dapat ditentukan laju pelayanan paket data berdasarkan persamaan 3.71 ditentukan :

$$\mu_{GGSN} = \frac{C_{GGSN}}{m} = \frac{10 \cdot 10^9}{1500 \times 8} = 833333,33 \text{ paket/s}$$

Untuk performansi sistem antrian, ditunjukkan dalam bentuk ρ (*utilization*), yang nilainya diasumsikan berubah dengan kenaikan tertentu. Dalam analisis skripsi ini diasumsikan bahwa nilai faktor utilisasi (ρ) diubah dari nilai 0,1 hingga 0,9 dengan kenaikan sebesar 0,1. Dari masing-masing nilai ρ dapat ditentukan laju kedatangan paket (λ) dan delay antrian (t_w) yang terjadi adalah:

$$\lambda_{GGSN} = \rho \times \mu_{GGSN} = 0,1 \times 833333,33 = 83333,33 \text{ paket/s}$$

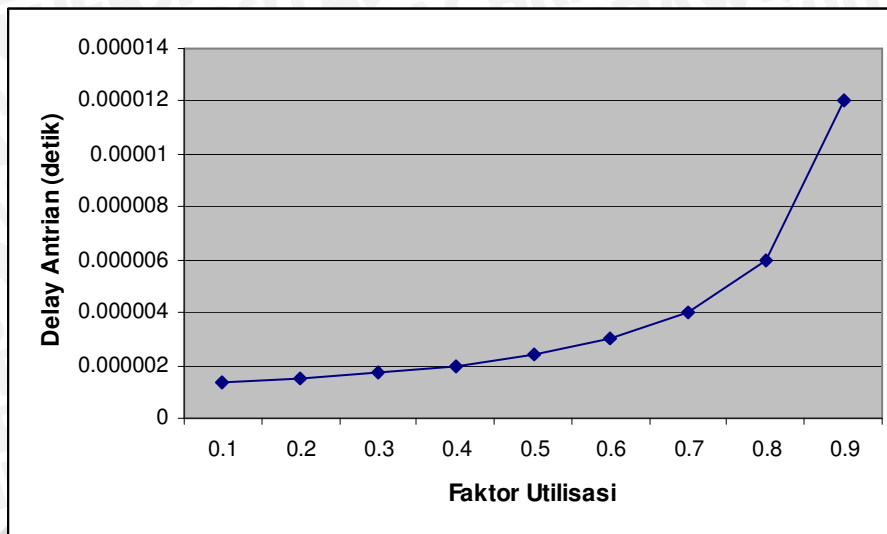
Delay antrian berdasarkan persamaan 3.76 adalah :

$$\begin{aligned} t_w &= \frac{\lambda_{GGSN} / \mu_{GGSN}}{\mu_{GGSN} - \lambda_{GGSN}} + \frac{1}{\mu_{GGSN}} \\ &= \frac{83333,33 / 833333,33}{833333,33 - 83333,33} + \frac{1}{833333,33} \\ &= 1,33 \cdot 10^{-5} \text{ det ik} \end{aligned}$$

Dengan cara yang sama, hasil perhitungan laju kedatangan paket (λ) dan *delay* antrian (t_w) untuk nilai faktor utilisasi (ρ) yang berubah-ubah dari 0,1 sampai 0,9 dapat ditunjukkan dalam Tabel 5.13.

Tabel 5.13 Hasil analisis *delay* antrian

No.	Faktor Utilisasi (ρ)	λ (paket/s)	<i>Delay</i> Antrian Total
1.	0,1	83333,333	$1,333 \cdot 10^{-6}$ s
2.	0,2	166666,666	$1,5 \cdot 10^{-6}$ s
3.	0,3	249999,999	$1,71429 \cdot 10^{-6}$ s
4.	0,4	333333,332	$2 \cdot 10^{-6}$ s
5.	0,5	416666,665	$2 \cdot 10^{-6}$ s
6.	0,6	499999,998	$3 \cdot 10^{-6}$ s
7.	0,7	583333,331	$4 \cdot 10^{-6}$ s
8.	0,8	666666,664	$6 \cdot 10^{-6}$ s
9.	0,9	749999,997	$1,2 \cdot 10^{-6}$ s



Gambar 5.11 Grafik hubungan faktor utilisasi dengan besarnya *delay* antrian pada jaringan UMTS

Dari hasil analisis *delay* antrian di atas maka dapat disimpulkan :

Semakin besar faktor utilisasi (ρ) yang digunakan maka laju kedatangan paket (λ) juga akan semakin besar sehingga mengakibatkan *delay* antrian (t_w) pada jaringan (GGSN) juga akan semakin besar.

Dari perhitungan *delay* proses (enkapsulasi/dekapsulasi), *delay* transmisi, *delay* propagasi dan *delay* antrian di atas maka dapat dihitung *delay* jaringan UMTS. Dengan menggunakan persamaan 3.1 diperoleh *delay* jaringan UMTS untuk *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps dan faktor utilisasi sebesar 0,1 yaitu :

$$\begin{aligned}
 t_{net} &= t_{proc} + t_w + t_T + t_p \\
 &= (3,335 \cdot 10^{-2} + 1,333 \cdot 10^{-6} + 2,316 \cdot 10^{-3} + 6,302 \cdot 10^{-5}) \text{ s} \\
 &= 3,573 \cdot 10^{-2} \text{ s} \\
 &= 35,73 \text{ ms}
 \end{aligned}$$

Dengan cara yang sama, hasil perhitungan *delay* jaringan UMTS untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte dengan mengubah faktor utilisasi dari 0,1 hingga 0,9 ditunjukkan dalam tabel 5.14, 5.15 dan 5.16.

Tabel 5.14 Hasil analisis *delay* jaringan

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay Jaringan		
			$\rho = 0,1$	$\rho = 0,2$	$\rho = 0,3$
1.	64 kbps	323,625 byte	0,035734273 s	0,03573444 s	0,035734654 s
2.	128 kbps	563,625 byte	0,051650186 s	0,051650353 s	0,051650567 s
3.	192 kbps	803,625 byte	0,06756873 s	0,067568897 s	0,067569111 s
4.	256 kbps	1043,625 byte	0,083979072 s	0,083979239 s	0,083979453 s
5.	320 kbps	1283,625 byte	0,101177018 s	0,101177185 s	0,101177399 s
6.	384 kbps	1523,625 byte	0,118625714 s	0,118625881 s	0,118626095 s

Sumber : perhitungan

Tabel 5.15 Hasil analisis *delay* jaringan

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay Jaringan		
			$\rho = 0,4$	$\rho = 0,5$	$\rho = 0,6$
1.	64 kbps	323,625 byte	0,03573494 s	0,03573534 s	0,03573594 s
2.	128 kbps	563,625 byte	0,051650853 s	0,051651253 s	0,051651853 s
3.	192 kbps	803,625 byte	0,067569397 s	0,067569797 s	0,067570397 s
4.	256 kbps	1043,625 byte	0,083979739 s	0,083980139 s	0,083980739 s
5.	320 kbps	1283,625 byte	0,101177685 s	0,101178085 s	0,101178685 s
6.	384 kbps	1523,625 byte	0,118626381s	0,118626781 s	0,118627381 s

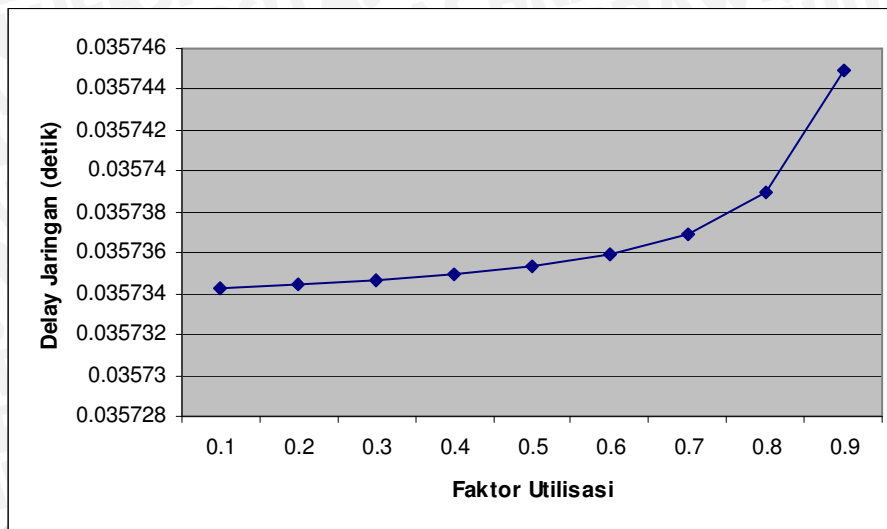
Sumber : perhitungan

Tabel 5.16 Hasil analisis *delay* jaringan

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay Jaringan		
			$\rho = 0,7$	$\rho = 0,8$	$\rho = 0,9$
1.	64 kbps	323,625 byte	0,03573694 s	0,03573894 s	0,03574494 s
2.	128 kbps	563,625 byte	0,051652853 s	0,051654853 s	0,051660853 s
3.	192 kbps	803,625 byte	0,067571397 s	0,067573397 s	0,067579397 s
4.	256 kbps	1043,625 byte	0,083981739 s	0,083983739 s	0,083989739 s
5.	320 kbps	1283,625 byte	0,101179685 s	0,101181685 s	0,101187685 s
6.	384 kbps	1523,625 byte	0,118628381 s	0,118630381 s	0,118636381 s

Sumber : perhitungan





Gambar 5.12 Grafik hubungan faktor utilisasi dengan besarnya *delay* jaringan untuk *bandwith* CODEC 64 Kbps pada jaringan UMTS

Dari seluruh nilai analisis *delay* yang didapatkan, maka nilai *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps dan faktor utilisasi sebesar 0,1 yaitu :

$$\begin{aligned}
 t_{tot} &= t_{CODEC} + t_{gk} + t_{net} \\
 &= (6 \cdot 10^{-2} + 1,0129 \cdot 10^{-8} + 3,573 \cdot 10^{-2}) \text{ s} \\
 &= 9,5734 \cdot 10^{-2} \text{ s} \\
 &= 95,734 \text{ ms}
 \end{aligned}$$

Dengan analisis yang sama, hasil perhitungan *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte dengan mengubah faktor utilisasi dari 0,1 hingga 0,9 ditunjukkan dalam tabel 5.17, 5.18 dan 5.19.

Tabel 5.17 Hasil analisis *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay end-to-end		
			$\rho = 0,1$	$\rho = 0,2$	$\rho = 0,3$
1.	64 kbps	323,625 byte	0,09573445 s	0,095734664 s	0,09573495 s
2.	128 kbps	563,625 byte	0,111650363 s	0,111650577 s	0,111650863 s
3.	192 kbps	803,625 byte	0,127568907 s	0,127569121 s	0,127569407 s

4.	256 kbps	1043,625 byte	0,143979249 s	0,143979463 s	0,143979749 s
5.	320 kbps	1283,625 byte	0,161177195 s	0,161177409 s	0,161177695 s
6.	384 kbps	1523,625 byte	0,178625891 s	0,178626105 s	0,178626391 s

Sumber : perhitungan

Tabel 5.18 Hasil analisis *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay end-to-end		
			$\rho = 0,4$	$\rho = 0,5$	$\rho = 0,6$
1.	64 kbps	323,625 byte	0,09573495 s	0,09573535 s	0,09573595 s
2.	128 kbps	563,625 byte	0,111650863 s	0,111651263 s	0,111651863 s
3.	192 kbps	803,625 byte	0,127569407 s	0,127569807 s	0,127570407 s
4.	256 kbps	1043,625 byte	0,143979749 s	0,143980149 s	0,143980749 s
5.	320 kbps	1283,625 byte	0,161177695 s	0,161178095 s	0,161178695 s
6.	384 kbps	1523,625 byte	0,178626391 s	0,178626791 s	0,178627391 s

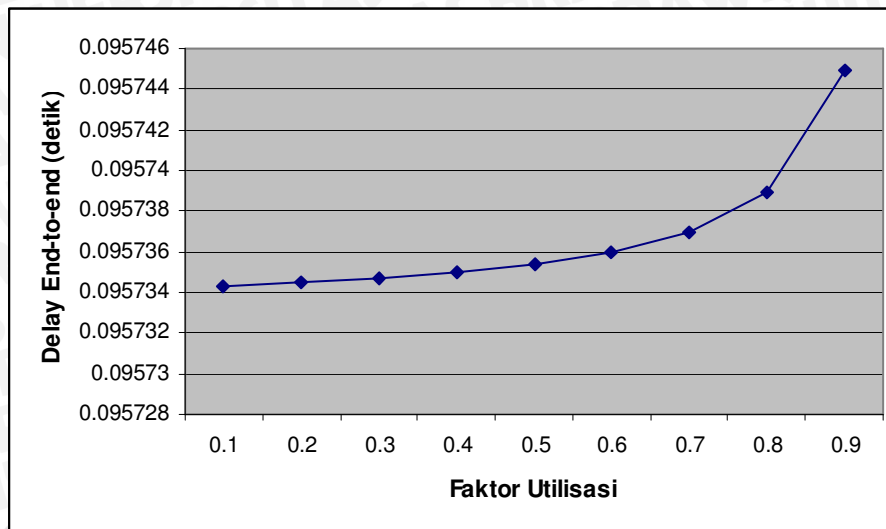
Sumber : perhitungan

Tabel 5.19 Hasil analisis *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Delay end-to-end		
			$\rho = 0,7$	$\rho = 0,8$	$\rho = 0,9$
1.	64 kbps	323,625 byte	0,09573695 s	0,09573895 s	0,09574495 s
2.	128 kbps	563,625 byte	0,111652863 s	0,111654863 s	0,111660863 s
3.	192 kbps	803,625 byte	0,127571407 s	0,127573407 s	0,127579407 s
4.	256 kbps	1043,625 byte	0,143981749 s	0,143983749 s	0,143989749 s
5.	320 kbps	1283,625 byte	0,161179695 s	0,161181695 s	0,161187695 s
6.	384 kbps	1523,625 byte	0,178628391 s	0,178630391 s	0,178636391 s

Sumber : perhitungan





Gambar 5.13 Grafik hubungan faktor utilisasi dengan besarnya *delay end-to-end* untuk *bandwidth* CODEC 64 Kbps

Dari hasil perhitungan dan grafik di atas menunjukkan bahwa :

- *Delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS dipengaruhi oleh besar panjang data untuk setiap *bandwidth CODEC video* yang digunakan serta faktor utilisasi yang digunakan. Semakin besar *bandwidth CODEC video* dan faktor utilisasi yang digunakan, maka semakin besar pula *delay end-to-end* penerapan aplikasi *videophone* pada jaringan UMTS.
- *Delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS terkecil yaitu 95,73445 ms dicapai pada saat menggunakan *bandwidth CODEC video* 64 kbps dengan panjang data 303,625 byte serta faktor utilisasi 0,1. Sedangkan *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS terbesar yaitu 178,636391 ms dicapai pada saat menggunakan *bandwidth CODEC video* 384 kbps dengan panjang data 1503,625 byte serta faktor utilisasi 0,9.
- *Delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS memenuhi standar *transfer delay* syarat layanan *Bearer IMT 2000* untuk aplikasi *real time* pada jaringan UMTS yaitu antara 20 – 300 ms.

5.3 Probabilitas *Packet Loss* Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS

5.3.1 Probabilitas *Packet Loss* Aplikasi *Videophone*

Dengan menggunakan persamaan 2.15 dan 2.16 dan $header_{UDP/RTP/IPv6}$ sebesar 60 byte (IPv6 40 byte, $header$ UDP 8 byte, $header$ RTP 12 byte) maka dapat dihitung probabilitas *packet loss* untuk aplikasi *videophone* untuk *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps yaitu :

$$\begin{aligned} \rho_{UDP/RTP/IP} &= P_{size} \cdot \rho_b \\ &= (60+323,625) \times 8 \times 10^{-7} \\ &= 3,07 \cdot 10^{-4} \end{aligned}$$

Dengan cara yang sama, hasil analisis probabilitas *packet loss* untuk aplikasi *videophone* untuk *bandwidth CODEC video* sebesar 128 kbps, 192 kbps, 256 kbps, 320 kbps, dan 384 kbps ditunjukkan pada Tabel 5.20.

Tabel 5.20 Hasil analisis probabilitas *packet loss* untuk aplikasi *videophone*

No.	<i>Bandwidth CODEC Video</i>	<i>Panjang Paket Videophone</i>	<i>Probabilitas Packet Loss Aplikasi Videophone</i>
1.	64 kbps	323,625 byte	$3,07 \cdot 10^{-4}$
2.	128 kbps	563,625 byte	$4,99 \cdot 10^{-4}$
3.	192 kbps	803,625 byte	$6,91 \cdot 10^{-4}$
4.	256 kbps	1043,625 byte	$8,83 \cdot 10^{-4}$
5.	320 kbps	1283,625 byte	$1,07 \cdot 10^{-3}$
6.	384 kbps	1523,625 byte	$1,27 \cdot 10^{-3}$

Sumber : Perhitungan

5.3.2 Probabilitas *Packet Loss* Jaringan UMTS

Probabilitas *packet loss* pada jaringan UMTS dihitung dari probabilitas *packet loss* yang terjadi pada *air interface (WCDMA air interface)* sistem UMTS, UTRAN (Node B, RNC) serta pada *core network (SGSN, GGSN)* dengan persamaan 3.77.

Dengan menggunakan persamaan 3.78 maka probabilitas *packet loss* pada UTRAN untuk *segment* data 323,625 byte dengan menggunakan *bandwidth CODEC video* 64 Kbps yaitu :

$$\rho_{UTRAN} = (l + l')Pe$$

$$\begin{aligned}
 &= (323,625 + 40) \times 8 \times 10^{-7} \\
 &= 363,625 \times 8 \times 10^{-7} \\
 &= 2,91.10^{-4}
 \end{aligned}$$

Sedangkan probabilitas *packet loss* pada *core network* dengan menggunakan persamaan 3.79 diperoleh :

$$\begin{aligned}
 \rho_{core\ network} &= (l + l')Pe \\
 &= (323,625 + 40) \times 8 \times 10^{-7} \\
 &= 363,625 \times 8 \times 10^{-7} \\
 &= 2,91.10^{-4}
 \end{aligned}$$

Probabilitas *packet loss* pada *air interface (WCDMA air interface)* jaringan UMTS baik untuk lintasan *forward* maupun lintasan *reverse* diperlukan parameter-parameter pada *link budget* pada tabel 3.4. Dengan menggunakan persamaan 3.80 – 3.90 diperoleh :

a. Lintasan Forward (MS ke BTS)

- EIRP (*Effective Isotropic Radiated Power*) *mobile station*

$$\begin{aligned}
 \text{EIRP}_{\text{MS}} &= P_{\text{TX-MS}} + G_{\text{MS}} - L_{\text{MS}} \\
 \text{EIRP}_{\text{MS}} &= 23 + 0 - 0 \\
 &= 23 \text{ dBm}
 \end{aligned}$$

- *Interference Margin*

Merupakan fungsi dari jumlah pembebanan pada *carrier* WCDMA.

$$\begin{aligned}
 I_m &= 10 \log \left(\frac{1}{1-\eta} \right) \\
 I_m &= 10 \log \left(\frac{1}{1-0,75} \right) = 6,02 \text{ dB}
 \end{aligned}$$

- Daya penerimaan *base station* tiap *user*

$$\begin{aligned}
 P_{\text{cu}} &= \text{EIRP}_{\text{MS}} - L_p + G_{\text{BS}} + L_f - I_m \\
 &= 23 - 138 + 18 + 2 - 6,02 \\
 &= -101,02 \text{ dBm}
 \end{aligned}$$

- *Thermal noise*

$$\begin{aligned}
 N_o &= 10 \log (290 \times 1,38 \cdot 10^{-23}) + NF_{\text{BS}} + 30 \\
 &= 10 \log (290 \times 1,38 \cdot 10^{-23}) + 7 + 30 \\
 &= -166,977 \text{ dBm/Hz}
 \end{aligned}$$

- *Energy per bit to noise density ratio*

$$\begin{aligned} E_b/N_o &= P_{cu} - 10 \log BR - 10 \log 10^{0,1N_o} \\ &= -101,02 - 10 \log 384000 - 10 \log 10^{0,1(-166,977)} \\ &= 10,117 \text{ dB} \\ &= 10,273 \end{aligned}$$

- Probabilitas bit error (P_e)

$$P_e = \frac{e^{-E_b/N_o}}{2\sqrt{\pi}(E_b/N_o)} = \frac{3,455 \times 10^{-5}}{11,359} = 3,04 \cdot 10^{-6}$$

- Dengan menggunakan persamaan 3.90 probabilitas *packet loss* pada *air interface* jaringan UMTS untuk lintasan *forward* (MS ke BTS) diperoleh :

$$\begin{aligned} \rho_{AI} &= 1 - (1 - P_e)^{l+l'} \\ &= 1 - (1 - 3,04 \cdot 10^{-6})^{2909} \\ &= 8,8 \cdot 10^{-3} \end{aligned}$$

b. Lintasan Reverse (BTS ke MS)

- Daya radiasi efektif kanal trafik

$$\begin{aligned} p_t &= P_t - 10 \log N_t - 10 \log C_f \\ &= 43 - 10 \log 76 - 10 \log 0,6 \\ &= 26,41 \text{ dBm} \end{aligned}$$

- Daya yang diterima kanal trafik

$$\begin{aligned} p_{tr} &= p_t - L_p + G_{MS} + L_{MS} - I_m \\ &= 26,41 - 138 + 0 + 0 - 6,02 \\ &= -99,61 \text{ dBm} \end{aligned}$$

- *Thermal noise*

$$\begin{aligned} N_o &= 10 \log (290 \times 1,38 \cdot 10^{-23}) + NF_{MS} + 30 \\ &= 10 \log (290 \times 1,38 \cdot 10^{-23}) + 10 + 30 \\ &= -163,977 \text{ dBm/Hz} \end{aligned}$$

- *Energy per bit to noise density ratio* kanal trafik

$$\begin{aligned} E_b/N_o &= p_{tr} - 10 \log BR - 10 \log 10^{0,1N_o} \\ &= -99,61 - 10 \log 384000 - 10 \log 10^{0,1(-163,977)} \\ &= 8,527 \text{ dB} \\ &= 7,123 \end{aligned}$$

- Probabilitas bit error (P_e)

$$Pe = \frac{e^{-Eb/No}}{2\sqrt{\pi}(Eb/No)} = \frac{8,063 \times 10^{-4}}{9,458} = 8,52 \cdot 10^{-5}$$

- Dengan menggunakan persamaan 3.90 probabilitas *packet loss* pada *air interface* jaringan UMTS untuk lintasan *reverse* (BTS ke MS) diperoleh :

$$\begin{aligned} \rho_{AI} &= 1 - (1 - Pe)^{h+l'} \\ &= 1 - (1 - 8,52 \cdot 10^{-5})^{2909} \\ &= 0,2195 \end{aligned}$$

Sehingga dengan menggunakan persamaan 3.77 diperoleh probabilitas *packet loss* pada jaringan UMTS untuk lintasan *forward* dan *reverse* yaitu :

$$\begin{aligned} \rho_{network (forward)} &= 1 - [(1 - \rho_{AI})(1 - \rho_{UTRAN})^{h1}(1 - \rho_{corework})^{h2}] \\ &= 1 - [(1 - 8,8 \cdot 10^{-3})(1 - 2,91 \cdot 10^{-4})^2(1 - 2,91 \cdot 10^{-4})^2] \\ &= 9,95 \cdot 10^{-3} \end{aligned}$$

$$\begin{aligned} \rho_{network (reverse)} &= 1 - [(1 - \rho_{AI})(1 - \rho_{UTRAN})^{h1}(1 - \rho_{corework})^{h2}] \\ &= 1 - [(1 - 0,2195)(1 - 2,91 \cdot 10^{-4})^2(1 - 2,91 \cdot 10^{-4})^2] \\ &= 2,2 \cdot 10^{-1} \end{aligned}$$

Probabilitas *packet loss* total untuk lintasan *forward* dan *reverse* diperoleh dengan menggunakan persamaan 2.14 yaitu :

$$\begin{aligned} \rho_{tot (forward)} &= 1 - [(1 - \rho_{network})(1 - \rho_{UDP/RTP/IP})] \\ &= 1 - [(1 - 9,95 \cdot 10^{-3})(1 - 3,07 \cdot 10^{-4})] \\ &= 1,03 \cdot 10^{-2} \end{aligned}$$

$$\begin{aligned} \rho_{tot (reverse)} &= 1 - [(1 - \rho_{network})(1 - \rho_{UDP/RTP/IP})] \\ &= 1 - [(1 - 2,2 \cdot 10^{-1})(1 - 3,07 \cdot 10^{-4})] \\ &= 2,21 \cdot 10^{-1} \end{aligned}$$

Dan probabilitas *packet loss* total *end-to-end* diperoleh :

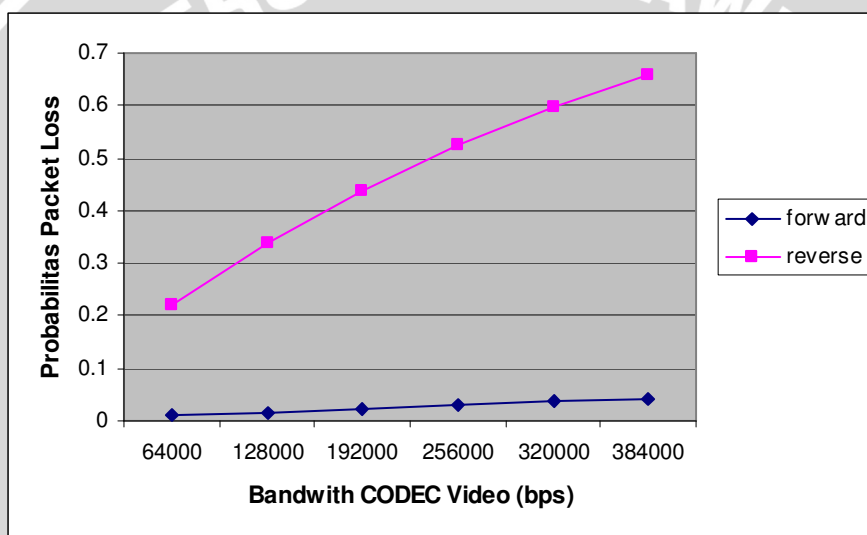
$$\begin{aligned} \rho_{tot (end-to-end)} &= \rho_{tot (forward)} + \rho_{tot (reverse)} \\ &= 1,03 \cdot 10^{-2} + 2,21 \cdot 10^{-1} \\ &= 0,231 \end{aligned}$$

Dengan cara yang sama, hasil analisis probabilitas *packet loss* total *end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *bandwidth CODEC video* sebesar 128 kbps, 192 kbps, 256 kbps, 320 kbps, dan 384 kbps ditunjukkan pada tabel 5.21.

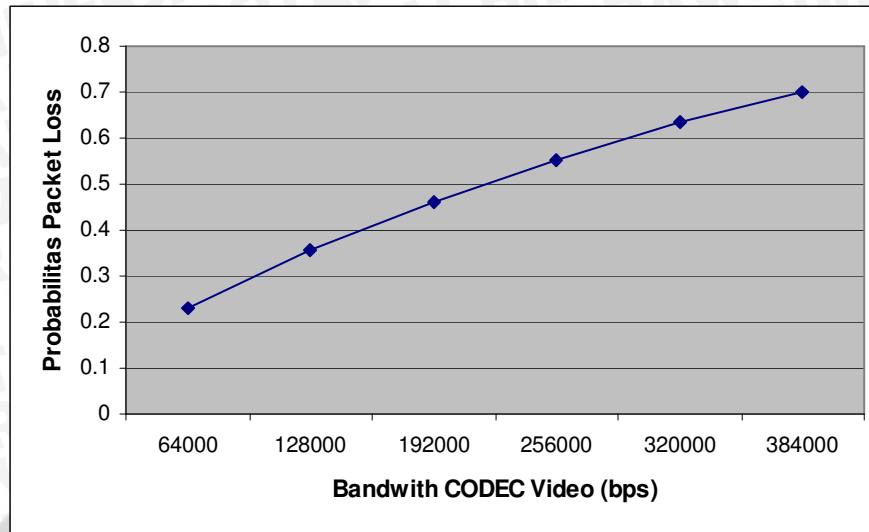
Tabel 5.21 Hasil analisis probabilitas *packet loss* total *end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Probabilitas Packet Loss Forward	Probabilitas Packet Loss Reverse	Probabilitas Packet Loss Total
1.	64 kbps	323,625 byte	0,0103	0,221	0,2313
2.	128 kbps	563,625 byte	0,017	0,339	0,356
3.	192 kbps	803,625 byte	0,0236	0,439	0,4626
4.	256 kbps	1043,625 byte	0,0302	0,524	0,5542
5.	320 kbps	1283,625 byte	0,0368	0,596	0,6328
6.	384 kbps	1523,625 byte	0,0433	0,658	0,7013

Sumber : Perhitungan



Gambar 5.14 Grafik hubungan *bandwidth CODEC video* dengan besarnya probabilitas *packet loss* total saat lintasan *forward* dan *reverse* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS



Gambar 5.15 Grafik hubungan *bandwith CODEC video* dengan besarnya probabilitas *packet loss total end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

Dari hasil analisis di atas dapat disimpulkan bahwa :

- Semakin besar konsumsi *bandwidth CODEC video* yang digunakan maka probabilitas *packet loss total end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS juga semakin besar.
- Probabilitas *packet loss total* untuk lintasan *forward* lebih kecil dibandingkan dengan probabilitas *packet loss total* untuk lintasan *reverse* yaitu sebesar $1,03 \cdot 10^{-2}$ untuk lintasan *forward* dan $2,21 \cdot 10^{-1}$ untuk lintasan *reverse* hal ini disebabkan karena probabilitas *packet loss* pada *air interface (WCDMA air interface)* jaringan UMTS untuk lintasan *forward* lebih kecil dibandingkan dengan lintasan *reverse*
- Probabilitas *packet loss total end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS terbesar yaitu 0,7013 dengan menggunakan *bandwidth CODEC video* sebesar 384 kbps. Sedangkan probabilitas *packet loss total end-to-end* terkecil adalah 0,2313 dengan menggunakan *bandwidth CODEC video* 64 kbps.

5.4 Throughput

Throughput merupakan parameter yang digunakan untuk mengetahui jumlah data yang diterima dalam keadaan baik terhadap waktu total transmisi

yang dibutuhkan dari sumber data ke penerima. Dalam skripsi ini *throughput* dihitung pada lintasan MT (*mobile terminal*) pengirim hingga MT (*mobile terminal*) sesuai dengan persamaan 3.92.

Besarnya nilai *throughput* untuk penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *segment* data 323,625 byte (2589 bit) dengan menggunakan *bandwidth CODEC video* 64 Kbps dan faktor utilisasi sebesar 0,1 diperoleh :

$$t_I = \frac{(PL_{paket} + H_{paket}) \times 8}{C_{trans}} = \frac{(42 + 6) \text{ byte} \times 8}{384000} = 0,001 \text{ detik}$$

Sedangkan waktu total yang dibutuhkan untuk mentransmisikan sebuah paket adalah :

$$t_T = \frac{t_{end-to-end}}{N_{paket}} = \frac{0,0957}{9,5} = 0,01 \text{ detik}$$

Sehingga didapatkan nilai konstanta α yaitu :

$$\alpha = \frac{t_T}{t_I} = \frac{0,01}{0,001} = 10$$

Sehingga didapatkan nilai *throughput* yaitu :

$$\begin{aligned} \lambda &= \frac{(1 - \rho_{tot})}{t_1 [1 + (\alpha - 1) \rho_{tot}]} \\ &= \frac{(1 - 0,2313)}{0,001 [1 + (10 - 1) 0,2313]} \\ &= 248 \text{ bps} \end{aligned}$$

Dengan analisis yang sama, hasil perhitungan *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk panjang paket 563,625 byte, 803,625 byte, 1043,625 byte, 1283,625 byte, dan 1523,625 byte dengan mengubah faktor utilisasi dari 0,1 hingga 0,9 ditunjukkan dalam tabel 5.22, 5.23 dan 5.24.

Tabel 5.22 Hasil analisis *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Throughput (bps)		
			$\rho = 0,1$	$\rho = 0,2$	$\rho = 0,3$
1.	64 kbps	2589 bit	248,0015125	248,0011879	248,0007704
2.	128 kbps	4509 bit	200,7260486	200,7258091	200,7255012

3.	192 kbps	6429 bit	163,7314129	163,731234	163,731004
4.	256 kbps	8349 bit	133,1791354	133,1790018	133,17883
5.	320 kbps	10269 bit	107,6276748	107,6275754	107,6274478
6.	384 kbps	12189 bit	86,07826089	86,07818749	86,07809311

Sumber : perhitungan

Tabel 5.23 Hasil analisis *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

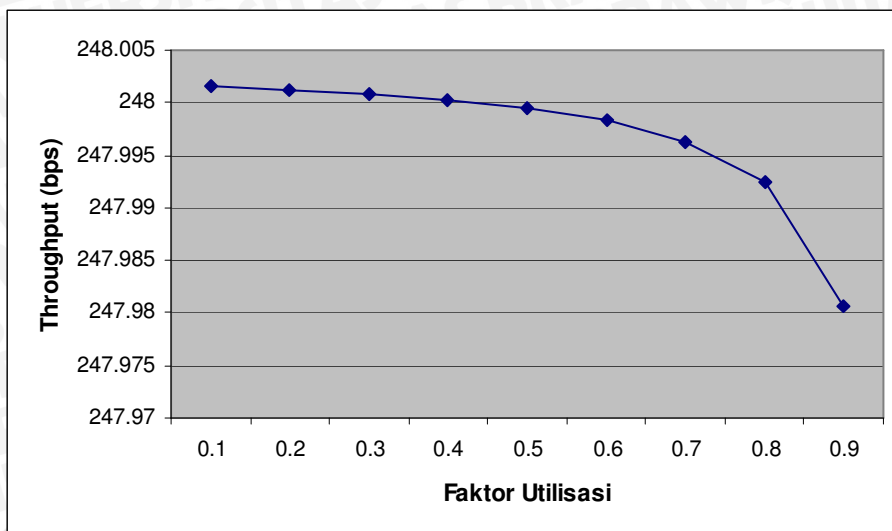
No.	Bandwidth CODEC Video	Panjang Paket Videophone	Throughput (bps)		
			$\rho = 0,4$	$\rho = 0,5$	$\rho = 0,6$
1.	64 kbps	2589 bit	248,0002138	247,9994346	247,9982658
2.	128 kbps	4509 bit	200,7250906	200,7245159	200,7236537
3.	192 kbps	6429 bit	163,7306974	163,730268	163,7296241
4.	256 kbps	8349 bit	133,1786009	133,1782802	133,1777991
5.	320 kbps	10269 bit	107,6272775	107,6270391	107,6266816
6.	384 kbps	12189 bit	86,07796728	86,07779112	86,07752687

Sumber : perhitungan

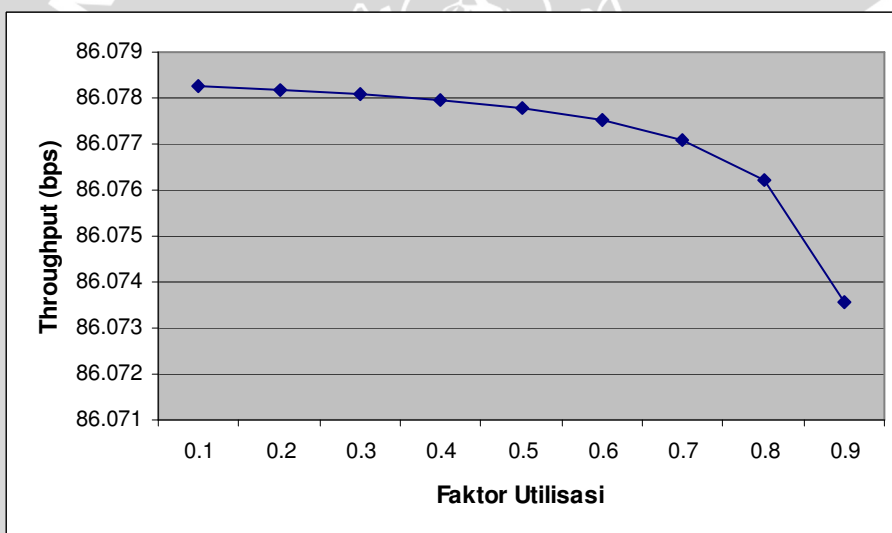
Tabel 5.24 Hasil analisis *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

No.	Bandwidth CODEC Video	Panjang Paket Videophone	Throughput (bps)		
			$\rho = 0,7$	$\rho = 0,8$	$\rho = 0,9$
1.	64 kbps	2589 bit	247,9963178	247,9924219	247,9807349
2.	128 kbps	4509 bit	200,7222168	200,7193431	200,7107224
3.	192 kbps	6429 bit	163,7285508	163,7264042	163,7199649
4.	256 kbps	8349 bit	133,1769973	133,1753938	133,1705835
5.	320 kbps	10269 bit	107,6260857	107,624894	107,621319
6.	384 kbps	12189 bit	86,07708647	86,07620568	86,07356342

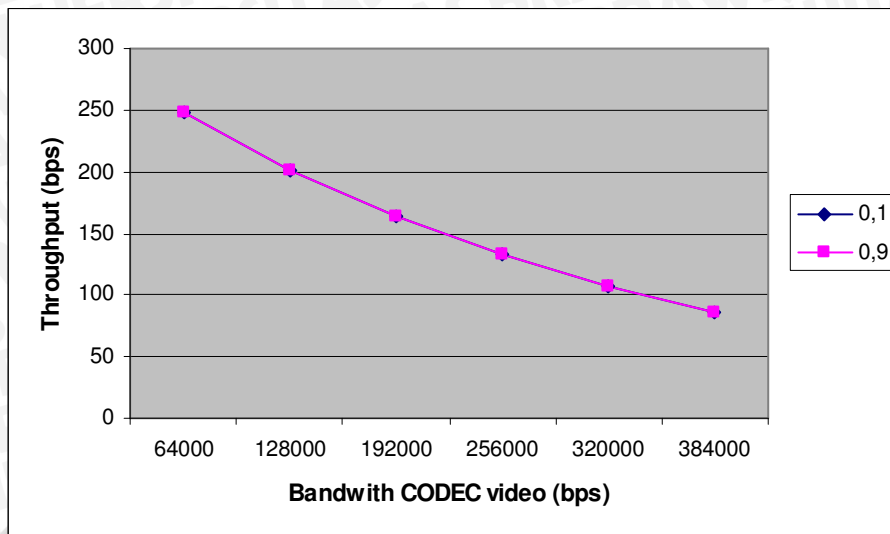
Sumber : perhitungan



Gambar 5.16 Grafik hubungan faktor utilisasi dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *bandwith CODEC 64 Kbps*



Gambar 5.17 Grafik hubungan faktor utilisasi dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk *bandwith CODEC 384 Kbps*



Gambar 5.18 Grafik hubungan *bandwith CODEC video* dengan nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS

Dari hasil analisis di atas dapat disimpulkan bahwa :

- Semakin besar *segment data* yang digunakan dengan faktor utilisasi yang sama maka nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS juga akan semakin turun, dengan *segment data* yang sama dan faktor utilisasi yang semakin naik nilai *throughput* juga akan semakin turun.
- Semakin besar konsumsi *bandwidth CODEC video* yang digunakan maka nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS akan semakin kecil.
- Nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS terbesar yaitu 248,0015125 bps, dicapai dengan menggunakan *segment data* sebesar 2589 bit dengan *bandwidth CODEC video* sebesar 64 kbps dan faktor utilisasi sebesar 0,1. Sedangkan nilai *throughput* terkecil yaitu 86,07356342 bps dicapai dengan menggunakan *segment data* sebesar 12189 bit dengan *bandwidth CODEC video* sebesar 384 kbps dan faktor utilisasi sebesar 0,9.

BAB VI

PENUTUP

6.1 Kesimpulan

Dari hasil analisis dan perhitungan tentang penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS diperoleh kesimpulan sebagai berikut :

1. Besar paket *videophone* dipengaruhi oleh *bandwith CODEC video* yang digunakan. Semakin besar *bandwith CODEC video* yang digunakan maka paket *videophone* juga semakin besar. Panjang paket aplikasi *videophone* terbesar yaitu 12189 bit atau 1523,625 byte dengan menggunakan *bandwith CODEC video* sebesar 384 Kbps. Sedangkan panjang paket aplikasi *videophone* terkecil yaitu 2589 bit atau 323,625 byte dengan menggunakan *bandwith CODEC video* sebesar 64 Kbps.
2. Nilai *delay end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS dipengaruhi oleh panjang data yang dikirimkan. Semakin besar data yang dikirimkan maka *delay end-to-end* akan semakin besar. Dengan mempertimbangkan *delay CODEC* aplikasi *videophone* , *delay gatekeeper* serta *delay* pada jaringan UMTS yang meliputi *delay* proses, *delay* transmisi, *delay* propagasi, *delay* antrian, serta faktor utilisasi maka diperoleh *delay end-to-end* terkecil yaitu 95,73445 ms dicapai pada saat menggunakan *bandwidth CODEC video* 64 kbps dengan panjang data 303,625 byte serta faktor utilisasi 0,1. Sedangkan *delay end-to-end* terbesar penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS yaitu 178,636391 ms dicapai pada saat menggunakan *bandwidth CODEC video* 384 kbps dengan panjang data 1503,625 byte serta faktor utilisasi 0,9.
3. Semakin besar konsumsi *bandwidth CODEC video* yang digunakan maka probabilitas *packet loss* total *end-to-end* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS juga semakin besar. Probabilitas *packet loss* total penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS untuk lintasan *forward* sebesar $1,03 \cdot 10^{-2}$ dan untuk lintasan *reverse* sebesar 0,221. Probabilitas *packet loss* total *end-to-end* penerapan IPv6

untuk aplikasi *videophone* pada jaringan UMTS terbesar yaitu 0,7013 dengan menggunakan *bandwidth CODEC video* sebesar 384 kbps. Sedangkan probabilitas *packet loss* total *end-to-end* terkecil adalah 0,2313 dengan menggunakan *bandwidth CODEC video* 64 kbps.

4. Nilai *throughput* penerapan IPv6 untuk aplikasi *videophone* pada jaringan UMTS dipengaruhi oleh panjang paket aplikasi *videophone*, besar *bandwidth CODEC video* aplikasi *videophone* yang digunakan, probabilitas *packet loss*, *delay end-to-end*, dan faktor utilisasi. Dengan mempertimbangkan parameter-parameter tersebut maka diperoleh nilai *throughput* terbesar yaitu 248,0015125 bps, dicapai dengan menggunakan *segment* data sebesar 2589 bit dengan *bandwidth CODEC video* sebesar 64 kbps dan faktor utilisasi sebesar 0,1. Sedangkan nilai *throughput* terkecil yaitu 86,07356342 bps dicapai dengan menggunakan *segment* data sebesar 12189 bit dengan *bandwidth CODEC video* sebesar 384 kbps dan faktor utilisasi sebesar 0,9.

6.2 Saran

Saran yang dapat diberikan berdasarkan analisis yang telah dilakukan dalam skripsi ini yaitu analisis dapat dikembangkan lebih lanjut dengan melakukan kajian untuk penerapan IPv6 pada jaringan UMTS dengan menggunakan aplikasi layanan yang berbeda.

DAFTAR PUSTAKA

- 3GPP. 2004, *UMTS, GPRS Service Description*, TS 23.060 version 5.8.0 Release 5.
- 3GPP. 2002, *Radio Interface Protocol Architecture*, TS 25.301 version 5.0.0 Release 5.
- 3GPP. 2004, *Medium Access Control (MAC) Protocol Specification*, TS 25.321 version 5.9.0 Release 5.
- 3GPP. 2004, *Radio Link Control (RLC) Protocol Specification*, TS 25.322 version 5.9.0 Release 5.
- 3GPP. 2005, *Packet Data Convergence Protocol (PDCP) Specification*, TS 25.323 version 5.5.0 Release 5.
- 3GPP. 2004, *GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface*, TS TS 29.060 version 5.9.0 Release 5.
- Anonymous, *Transition to IPv6 in 2G and 3G mobile networks*, (<http://www.nokia.com/whitepaper>)
- Anonymous, *Cisco Gateway GPRS Support Node*, (www.cisco.com)
- Anonymous, *UMTS Protocols and Protocol Testing*, (<http://www.iec.org>)
- IEEE Metrocon. 2001, *A Comparative Study UMTS (WCDMA) and CDMA 2000 Networks*, (www.awardsolutions.com)
- Schwartz, Mischa. 1987. *Telecommunication Networks : Protocols, Modelling and Analysis*. USA : Addison-Wesley Publishing Company.
- Heywood, Drew. 2001, *Konsep & Penerapan Microsoft TCP/IP*, Yogyakarta: Andi.
- Purbo, Onno W. 2002, *Buku Pintar Internet TCP/IP*, Jakarta: Elex Media Komputindo.
- Stallings, W. 2001, *Dasar-dasar Komunikasi Data*, Jakarta: Salemba Teknika.
- Subchi, M. 2005, *Penerapan Aplikasi Videophone Pada Sistem Komunikasi Bergerak Selular CDMA 200-1x*, Malang: Jurusan Teknik Elektro FT. Unibraw.
- Syarvianti, E. 2005, *Kajian Penerapan IPv6 pada Jaringan TCP/IP*, Malang: Jurusan Teknik Elektro FT. Unibraw.

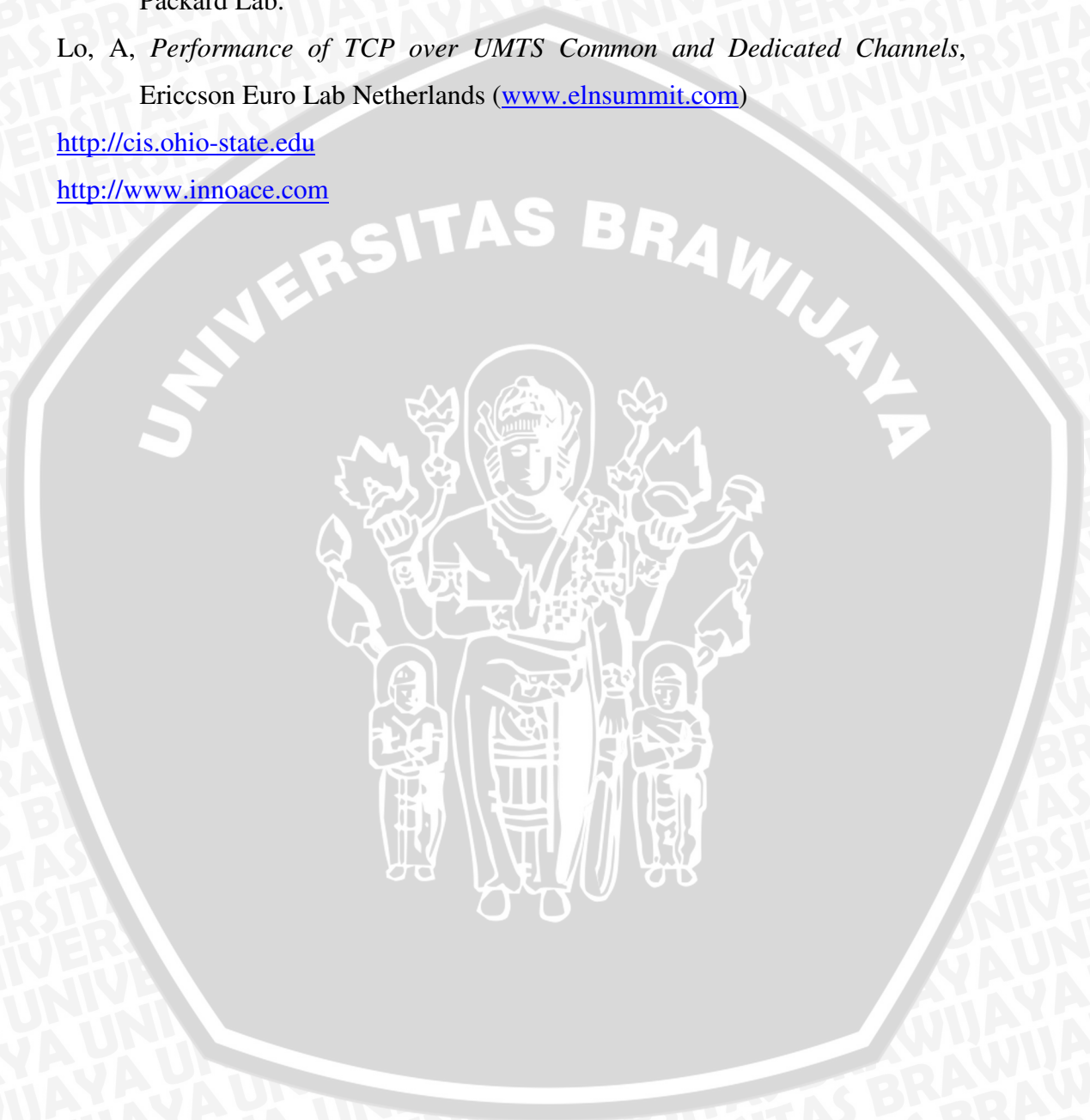
Prihandoko, A. 2005, *Perencanaan Sistem UTRA FDD (UMTS Terrestrial Radio Access Frequency Division Duplex pada Jaringan UMTS*, Malang: Jurusan Teknik Elektro FT. Unibraw.

Apostolopoulos, John G, *Video Communication and Video Streaming*, Hewlett-Packard Lab.

Lo, A, *Performance of TCP over UMTS Common and Dedicated Channels*, Ericsson Euro Lab Netherlands (www.elnsummit.com)

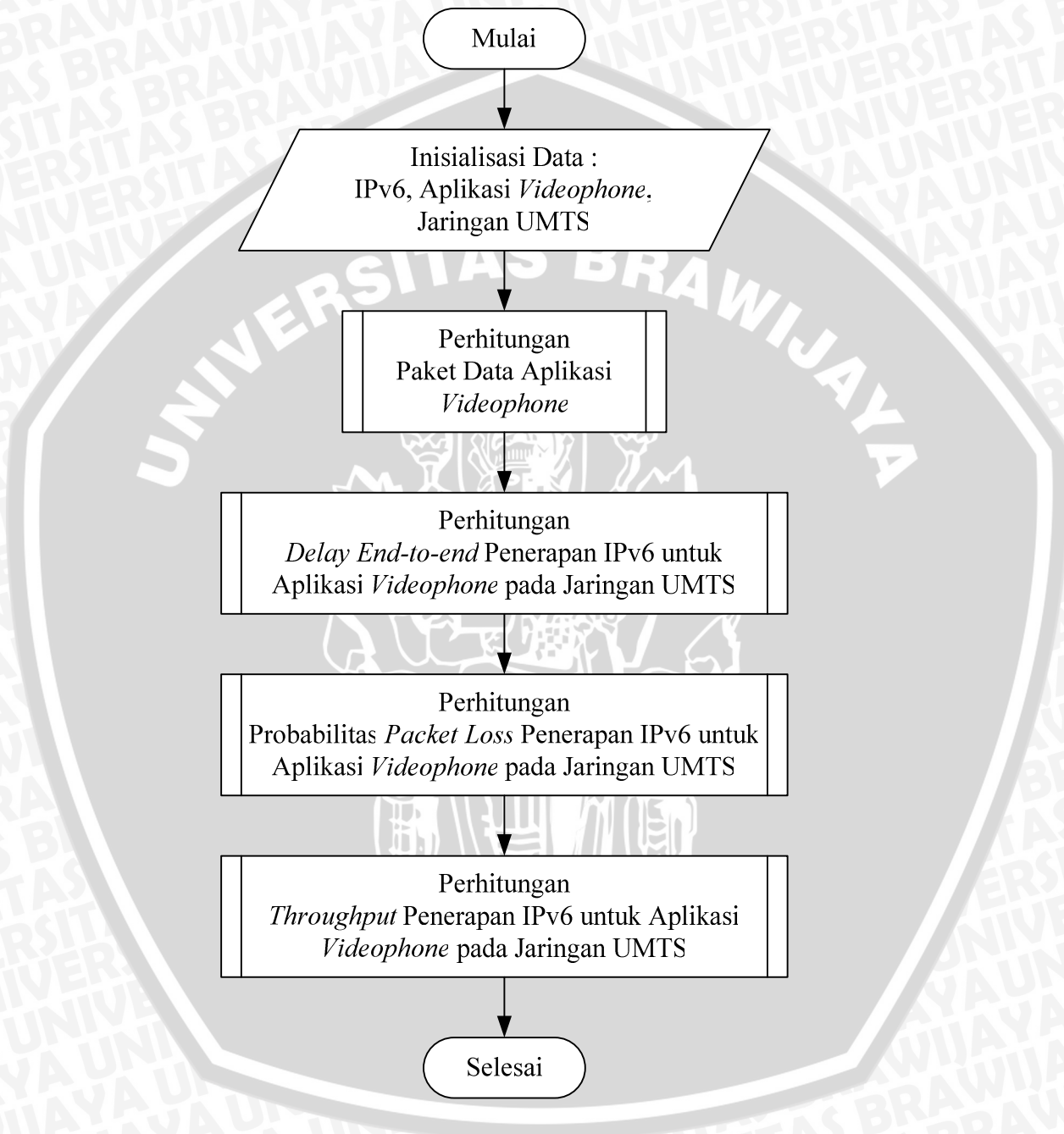
<http://cis.ohio-state.edu>

<http://www.innoace.com>



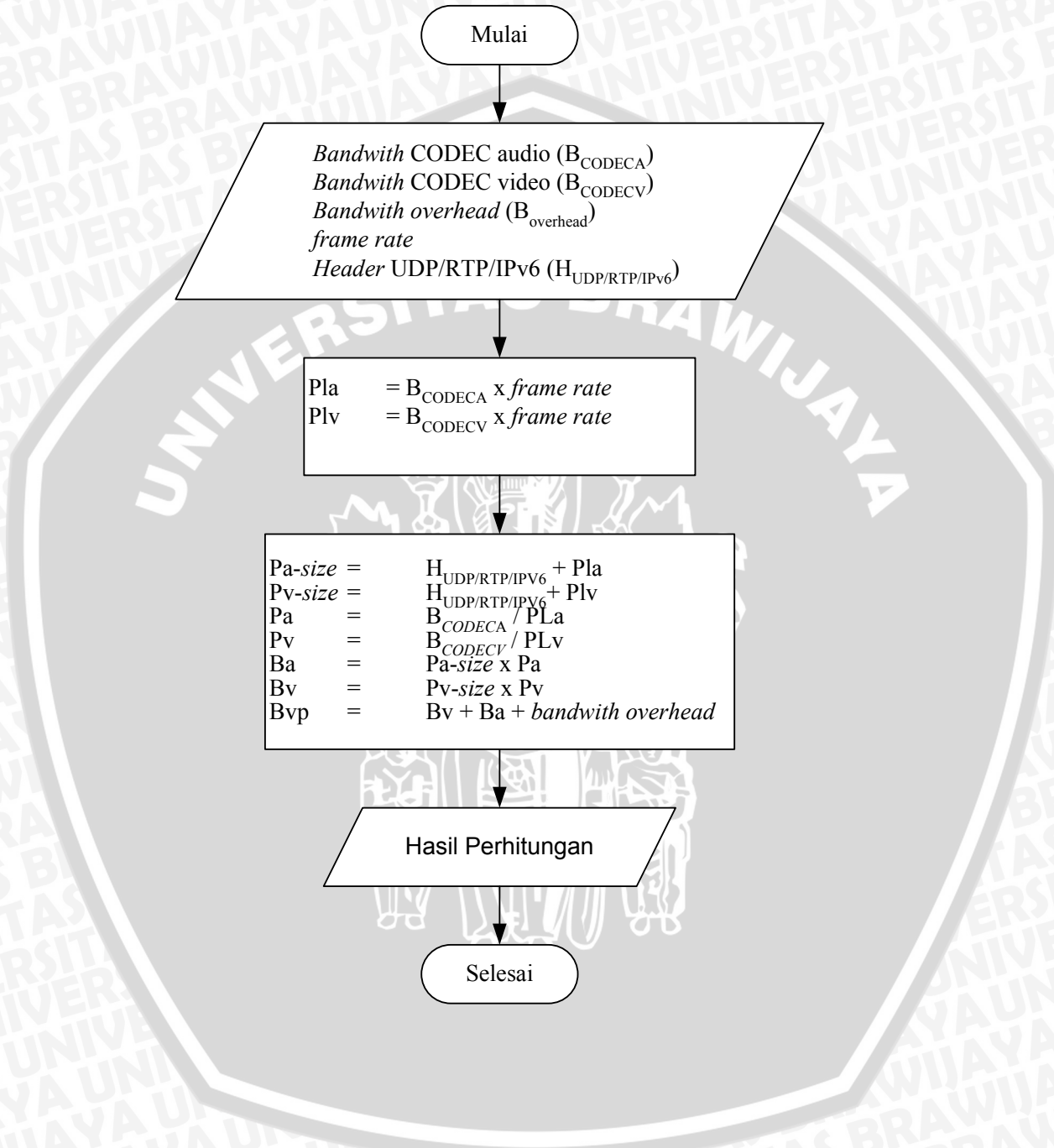
LAMPIRAN 1

Diagram Alir Analisis Penerapan IPv6 untuk Aplikasi *Videophone* pada Jaringan UMTS



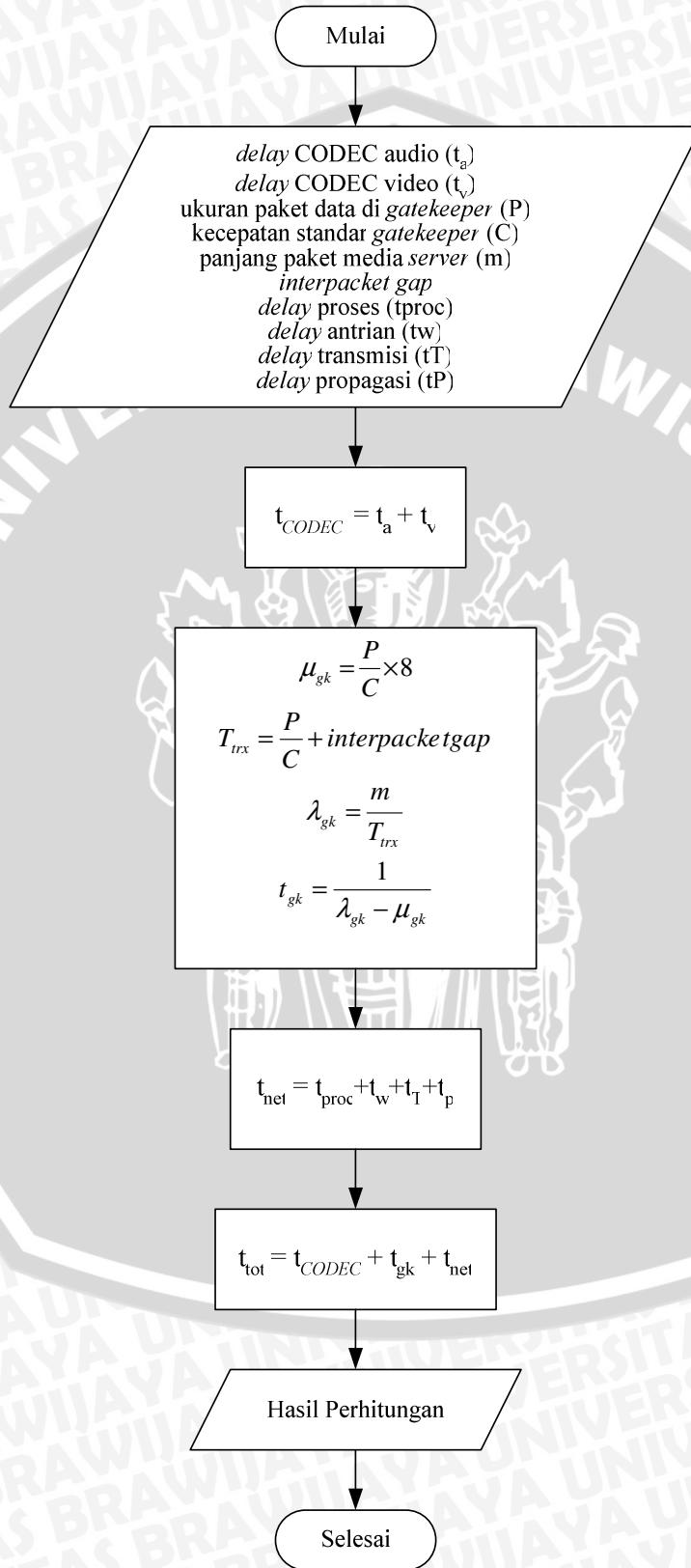
LAMPIRAN 2

Diagram Alir Perhitungan Paket Data *Videophone*



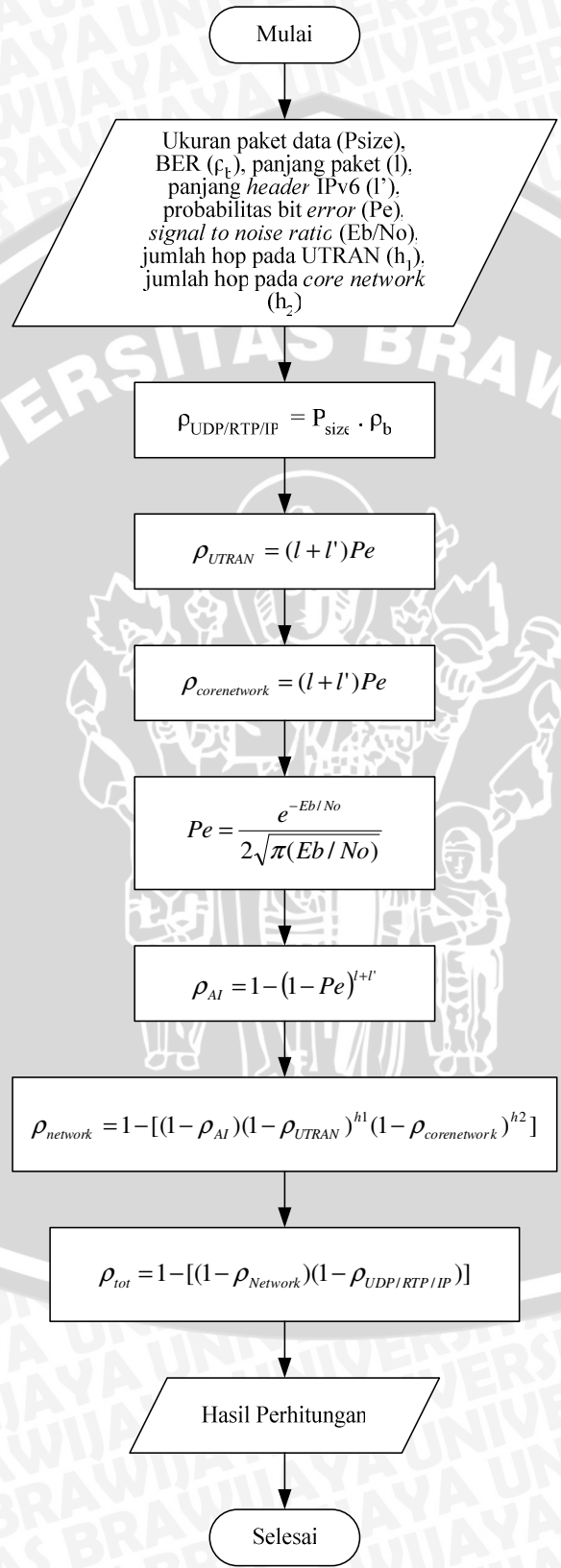
LAMPIRAN 3

Diagram Alir Perhitungan *Delay End-to-End*



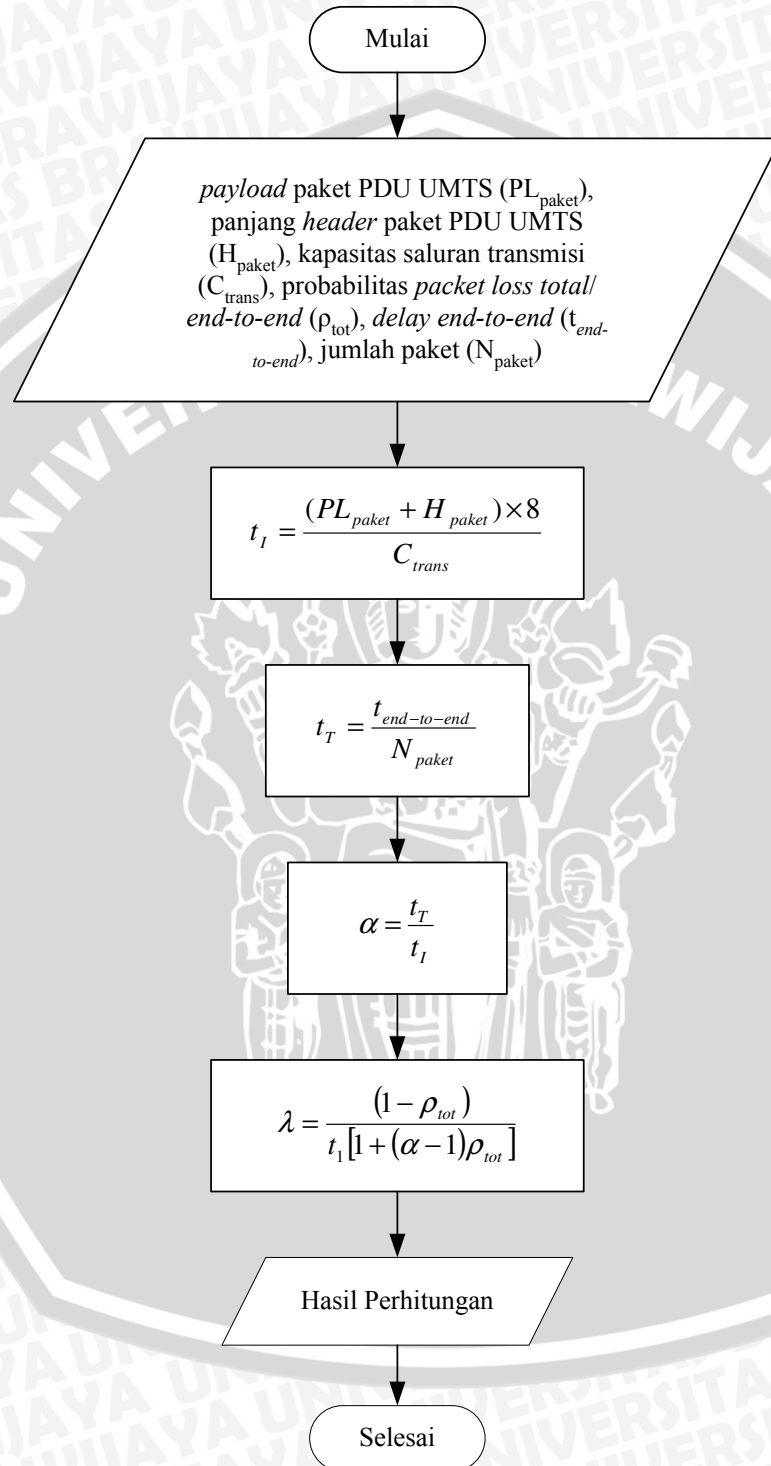
LAMPIRAN 4

Diagram Alir Perhitungan Probabilitas *Packet Loss*



LAMPIRAN 5

Diagram Alir Perhitungan *Throughput*



LAMPIRAN 6 Hasil analisis delay end-to-end IPv6

faktor utilisasi	t CODEC	t_gk	tnet 64 kbps	t end-to-end	tnet 128 kbps	t end-to-end	tnet 192 kbps	t end-to-end
0,1	0,06	1,0129E-08	0,035734273	0,095734283	0,051650186	0,111650196	0,06756873	0,12756874
0,2	0,06	1,0129E-08	0,03573444	0,09573445	0,051650353	0,111650363	0,067568897	0,127568907
0,3	0,06	1,0129E-08	0,035734654	0,095734664	0,051650567	0,111650577	0,067569111	0,127569121
0,4	0,06	1,0129E-08	0,03573494	0,09573495	0,051650853	0,111650863	0,067569397	0,127569407
0,5	0,06	1,0129E-08	0,03573534	0,09573535	0,051651253	0,111651263	0,067569797	0,127569807
0,6	0,06	1,0129E-08	0,03573594	0,09573595	0,051651853	0,111651863	0,067570397	0,127570407
0,7	0,06	1,0129E-08	0,03573694	0,09573695	0,051652853	0,111652863	0,067571397	0,127571407
0,8	0,06	1,0129E-08	0,03573894	0,09573895	0,051654853	0,111654863	0,067573397	0,127573407
0,9	0,06	1,0129E-08	0,03574494	0,09574495	0,051660853	0,111660863	0,067579397	0,127579407
			tnet 256 kbps	t end-to-end	tnet 320 kbps	t end-to-end	tnet 384 kbps	t end-to-end
			0,083979072	0,143979082	0,101177018	0,161177028	0,118625714	0,178625724
			0,083979239	0,143979249	0,101177185	0,161177195	0,118625881	0,178625891
			0,083979453	0,143979463	0,101177399	0,161177409	0,118626095	0,178626105
			0,083979739	0,143979749	0,101177685	0,161177695	0,118626381	0,178626391
			0,083980139	0,143980149	0,101178085	0,161178095	0,118626781	0,178626791
			0,083980739	0,143980749	0,101178685	0,161178695	0,118627381	0,178627391
			0,083981739	0,143981749	0,101179685	0,161179695	0,118628381	0,178628391
			0,083983739	0,143983749	0,101181685	0,161181695	0,118630381	0,178630391
			0,083989739	0,143989749	0,101187685	0,161187695	0,118636381	0,178636391

LAMPIRAN 7 Hasil analisis delay end-to-end IPv4

faktor utilisasi	t CODEC	t gk	tnet 64 kbps	t checksum	t end-to-end	tnet 128 kbps	t checksum	t end-to-end
0,1	0,06	1,0129E-08	0,030639917	0,0418	0,132439927	0,04655583	0,0603	0,16685584
0,2	0,06	1,0129E-08	0,030640084	0,0418	0,132440094	0,046555997	0,0603	0,166856007
0,3	0,06	1,0129E-08	0,030640298	0,0418	0,132440308	0,046556211	0,0603	0,166856221
0,4	0,06	1,0129E-08	0,030640584	0,0418	0,132440594	0,046556497	0,0603	0,166856507
0,5	0,06	1,0129E-08	0,030640984	0,0418	0,132440994	0,046556897	0,0603	0,166856907
0,6	0,06	1,0129E-08	0,030641584	0,0418	0,132441594	0,046557497	0,0603	0,166857507
0,7	0,06	1,0129E-08	0,030642584	0,0418	0,132442594	0,046558497	0,0603	0,166858507
0,8	0,06	1,0129E-08	0,030644584	0,0418	0,132444594	0,046560497	0,0603	0,166860507
0,9	0,06	1,0129E-08	0,030650584	0,0418	0,132450594	0,046566497	0,0603	0,166866507
			tnet 192 kbps	t checksum	t end-to-end	tnet 256 kbps	t checksum	t end-to-end
			0,062471713	0,0787	0,201171723	0,07849312	0,0973	0,23579313
			0,06247188	0,0787	0,20117189	0,078493287	0,0973	0,235793297
			0,062472094	0,0787	0,201172104	0,078493501	0,0973	0,235793511
			0,06247238	0,0787	0,20117239	0,078493787	0,0973	0,235793797
			0,06247278	0,0787	0,20117279	0,078494187	0,0973	0,235794197
			0,06247338	0,0787	0,20117339	0,078494787	0,0973	0,235794797
			0,06247438	0,0787	0,20117439	0,078495787	0,0973	0,235795797
			0,06247638	0,0787	0,20117639	0,078497787	0,0973	0,235797797
			0,06248238	0,0787	0,20118239	0,078503787	0,0973	0,235803797
			tnet 320 kbps	t checksum	t end-to-end	tnet 384 kbps	t checksum	t end-to-end
			0,095190661	0,117	0,272190671	0,112083659	0,138	0,310083669
			0,095190828	0,117	0,272190838	0,112083826	0,138	0,310083836
			0,095191042	0,117	0,272191052	0,11208404	0,138	0,31008405
			0,095191328	0,117	0,272191338	0,112084326	0,138	0,310084336
			0,095191728	0,117	0,272191738	0,112084726	0,138	0,310084736
			0,095192328	0,117	0,272192338	0,112085326	0,138	0,310085336
			0,095193328	0,117	0,272193338	0,112086326	0,138	0,310086336
			0,095195328	0,117	0,272195338	0,112088326	0,138	0,310088336
			0,095201328	0,117	0,272201338	0,112094326	0,138	0,310094336

CISCO GGSN HARDWARE PLATFORMS

The Cisco GGSN is supported on two standard Cisco routing platforms, the high-end Cisco 7600 Series Router and the market-entry Cisco 7200 Series VXR routers. These platforms are unchanged from the normal routing platform except they require special Cisco IOS Software to run the Cisco GGSN function. An overview of the hardware platforms follows (refer to Figure 1).

The Cisco 7600 Series Router is a high-performance router deployed at the network edge, where performance, IP services, redundancy, and fault resiliency are critical. Combined with a central route processor and forwarding engine, the Cisco 7600 Series provides 30-Mpps forwarding rates and up to 256 Gbps of total throughput.

The versatile Cisco 7600 Series system scales WAN connectivity from OC-48/STM-16 to DS-0 and LAN connectivity from 10 Gigabit Ethernet through 10-Mbps Ethernet. The Cisco 7600 Series delivers these capabilities while implementing high-touch, hardware-accelerated IP services through the Cisco Parallel Express Forwarding (PXF) network processor.

Figure 1
Cisco 7600 Series Router



The Cisco 7600 Series delivers up to 13 slots of optical LAN, WAN, and metropolitan-area network (MAN) networking at the network edge, helping enable service providers to offer high-value, Differentiated Services. It provides mobile service providers with the ability to deploy the advanced network infrastructure necessary to succeed in demanding, high-traffic environments. The Cisco 7600 Series enhanced chassis delivers design improvements that incorporate redundant, tiered-speed fan trays with configurable options for route processor, switch fabric, and power supply redundancy.

The Cisco 7600 Series routers accommodate a broad selection of line cards supporting numerous applications, including:

- Optical services modules (OSMs):
 - OC-3/STM-1
 - OC-12/STM-4
 - OC-48/ STM-16 packet over SONET (POS)

- OC-12/STM-4 ATM
- Gigabit Ethernet WAN
- Channelized T3 (CT3)
- OC-12/STM-4
- FlexWAN module: Supports Cisco 7200 Series and Cisco 7500 Series WAN port adapters for DS-0 to OC-3
- LAN Ethernet modules:
 - 10/100 Mbps
 - Gigabit Ethernet
 - 10 Gigabit Ethernet
- Services modules:
 - Firewall, network analysis, content switching, and Secure Sockets Layer (SSL), providing the foundation for a powerful combination of speed and services

The Cisco 7600 Series Router is an outstanding choice for multiple applications. When combined with the Cisco 7600 Multiprocessor WAN Application Module (MWAM) and Cisco GGSN software, it delivers a high-performance, highly flexible, redundant platform for mobile service delivery.

For more information about the Cisco 7600 Series, visit <http://www.cisco.com/go/7600>.

Cisco GGSN MWAM

The Cisco MWAM is a Cisco IOS Software application module that can be installed in the Cisco Catalyst® 6500 Series Switch or the Cisco 7600 Series Router (refer to Figure 2). The module runs multiple instances of a Cisco application; for example, the Cisco PDSN, the Cisco GGSN, or the Cisco Service Selection Gateway (SSG).

Figure 2
Cisco MWAM Card



Processor Complexes

A Cisco MWAM can run five instances of Cisco IOS Software. Each Cisco MWAM has one Cisco IOS application image—all processors on the MWAM are loaded with the same image. Mixed applications (for example, PDSN, GGSN, and SSG) on the same MWAM are not supported. The Cisco Catalyst 6500 Series chassis and Cisco 7600 Series chassis can accommodate up to four Cisco GGSN MWAMs and additional cards. Therefore, multiple applications could run in the same chassis on different MWAMs.