

## BAB IV

### HASIL DAN PEMBAHASAN



#### 4.1 Konfigurasi Jaringan VoIP pada *mobile* WiMAX

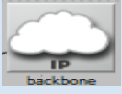


Dalam konfigurasi jaringan VoIP pada *mobile* WiMAX akan ditentukan komponen komponen jaringan, karakteristik trafik yang akan dibangkitkan serta penentuan parameter parameter simulasi yang diimplementasikan dalam model jaringan yang akan dirancang untuk mengetahui performansi VoIP pada *mobile* WiMAX.

##### 4.1.1 Komponen Jaringan

Pada simulasi VoIP melalui *mobile* WiMAX pada OPNET Modeler diperlukan model-model jaringan yang merepresentasikan dari kondisi jaringan sebenarnya yang disebut dengan *node model*. Dalam *node model* tersebut terdapat konfigurasi yang disesuaikan dengan standar *mobile* WiMAX 802.16e. Pada tabel 4.1 berikut ini akan diberikan penjelasan mengenai komponen-komponen jaringan (*node models*) yang akan digunakan dalam simulasi menggunakan OPNET Modeler 14.5

**Tabel 4.1** Komponen-komponen jaringan yang digunakan dalam simulasi

Nama	Fungsi	Simbol
Wimax_bs_ethernet4_slip4_router	Untuk mempresentasikan <i>Base Station</i> (BS) pada WiMAX	
Wimax_ss_wkstn	Untuk mempresentasikan <i>Mobile Station</i> (MS) sebagai sumber dan tujuan dengan menggunakan aplikasi TCPIP	

Nama	Fungsi	Simbol
Router_slip64_dc	Untuk mempresentasikan <i>gateway</i> yang menghubungkan dengan jaringan internet	
Wimax Config	Untuk melakukan pengaturan konfigurasi parameter-parameter jaringan WiMAX	
Server	Untuk mengatur lalu lintas data yang terjadi pada jaringan sehingga dapat saling terhubung	

#### 4.1.1 Parameter-Parameter Jaringan WiMAX

Dalam simulasi ini digunakan pula parameter-parameter jaringan *mobile* WiMAX 802.16e yang ditentukan pada konfigurasi model-model dalam simulasi sehingga diperoleh hasil simulasi yang mendekati kondisi nyata. Pada Tabel 4.2 menunjukkan data-data yang merupakan parameter-parameter jaringan WiMAX yang digunakan dalam simulasi ini. Parameter dalam tabel 4.2 akan diimplementasikan ke dalam model model yang telah ditentukan sebelumnya, yaitu pada model *Wimax\_bs\_ethernet4\_slip4\_router*, *Wimax\_ss\_wkstn* dan *Wimax Config*.

**Tabel 4.2** Parameter-parameter *mobile* WiMAX 802.16e dalam simulasi

Parameter	Nilai
Frekuensi Kerja	2,5 GHz
Metode Transmisi	TDD
Teknik Transmisi	OFDMA
Bandwidth Kanal	10 MHz
Jumlah Subcarrier	1024
Teknik Modulasi	64-QAM <sup>3</sup> / <sub>4</sub>
Tinggi BS	32 meter
Gain Antena BS	15 dBi
Tinggi MS	1,5 meter
Daya Maksimum MS	23 dBm
Gain Antena MS	-1 dBi

(Sumber : WiMAX Forum, 2006)

#### 4.1 Desain Simulasi

Simulasi performansi VoIP melalui *mobile* WiMAX dilakukan dengan menggunakan OPNET Modeler 14.5 dimana perlu dilakukan konfigurasi jaringan WiMAX kedalam *network* model simulasi.

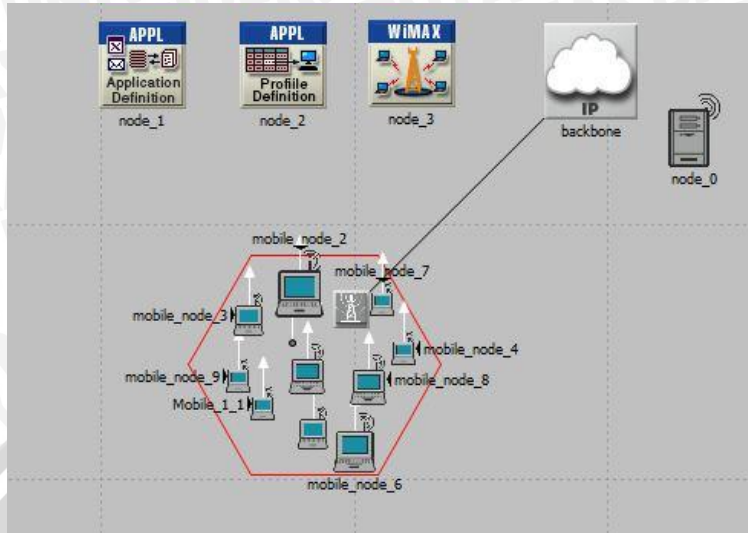
#### 4.2.1. Topologi Jaringan

Pada penelitian ini dalam membangun topologi jaringan WiMAX pada OPNET Modeler 14.5 digunakan menu *Wireless Network Deployment* (WND). Menu tersebut memberikan kemudahan dalam membangun dan melakukan konfigurasi jaringan *wireless*. Dalam pembuatan topologi jaringan WiMAX menggunakan menu WND diperlukan pula beberapa parameter yang digunakan seperti yang ditunjukkan pada tabel 4.3.

**Table 4.3** Parameter dalam pembuatan topologi jaringan WiMAX

Parameter	Nilai
Daya Maksimum BS	43 dBm
Daya Maksimum MS	15 dBi
Model <i>Pathloss</i> dan <i>Multipath</i>	Pedestriaan
Jumlah <i>Cell</i>	1
<i>Cell</i> Radius	2 km
Bentuk <i>Cell</i>	<i>Hexagon</i>
Penempatan MS	Random
Kecepatan MS	0.5 m/s
Area Pergerakan <i>Cell</i>	Dalam <i>Cell</i>
Ketinggian MS	1.5 m
Mobilitas MS	Random

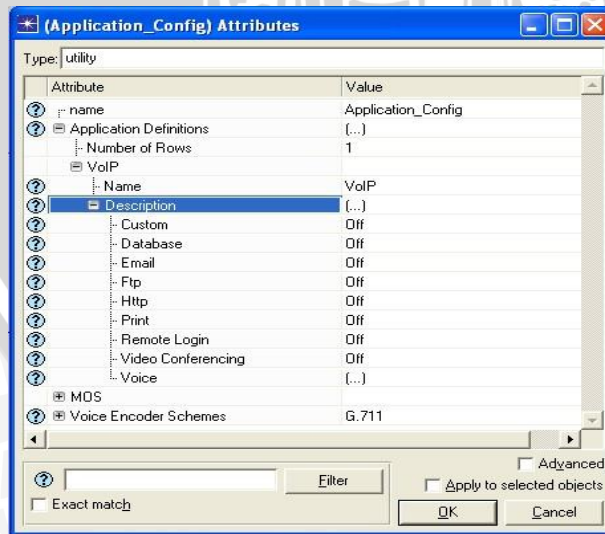
Setelah parameter yang ditentukan terimplementasikan maka perlu ditambahkan dengan beberapa node yaitu Profile Config, Application Config, dan Wimax Config. Sehingga terbentuklah topologi jaringan *mobile* WiMAX seperti Gambar 4.1.



Gambar 4.1 Topologi jaringan WiMAX dalam simulasi

#### 4.2.2. Implementasi Layanan VoIP Pada Jaringan Mobile WiMAX

Setelah membuat topologi jaringan *mobile* WiMAX maka perlu dilakukan pengimplementasian layanan. Pada simulasi ini jenis aplikasi yang akan digunakan adalah layanan VoIP dengan memanfaatkan jaringan WiMAX. Untuk mengimplementasikan aplikasi tersebut dalam OPNET Modeler, digunakan *node model* Application Config. Dalam *node model* tersebut terdapat aplikasi-aplikasi standar yang dapat disesuaikan dengan kebutuhan. Gambar 4.2 menunjukkan konfigurasi pada *node model* Application Config.



Gambar 4.2 Konfigurasi pada atribut model Application Config

Konfigurasi parameter VoIP dirancang dengan menentukan tabel *voice*. Parameter yang digunakan dalam layanan untuk aplikasi VoIP dalam simulasi ini adalah :

- *audio codec* yang digunakan adalah standar G.711 dengan *bit rate* 64 Kbps dan *frame size* 10 ms
- protokol yang digunakan adalah H.323

Pada simulasi ini codec yang digunakan diubah ubah untuk melihat pengaruhnya terhadap penggunaan variasi *service class*. Codec yang digunakan G.711, G.723 dan G.729 dapat diubah pada tabel *voice* yang terdapat pada *application definitions*.

Setelah konfigurasi pada *node model* Application Config, perlu dilakukan konfigurasi pada Profile Config untuk menentukan perilaku *user* pada layanan yang akan dirancang. Perilaku *user* yang telah ditentukan, diimplementasikan ke dalam *node model* Profile Config yang akan digunakan selama proses simulasi berlangsung, sebagaimana ditunjukkan dalam Gambar 4.3 berikut:

Attribute	Value
name	node_2
Profile Configuration	(...)
Number of Rows	1
wimax	
Profile Name	wimax
Applications	(...)
Number of Rows	1
voice	
Name	voice
Start Time Offset (seconds)	uniform (5,10)
Duration (seconds)	End of Profile
Repeatability	Unlimited
Operation Mode	Simultaneous
Start Time (seconds)	uniform (100,110)
Duration (seconds)	End of Simulation
Repeatability	Once at Start Time

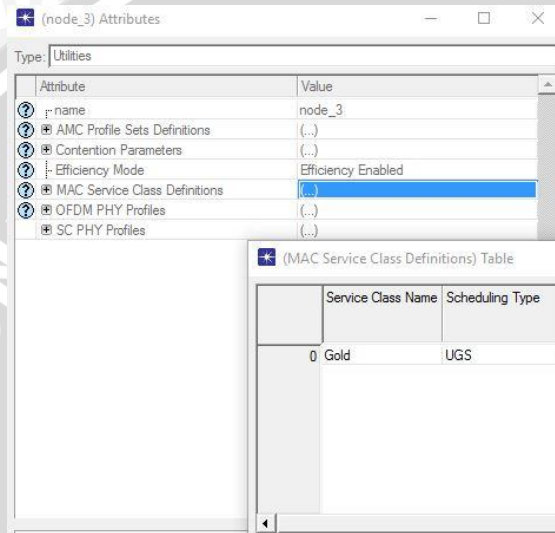
Gambar 4.3 Konfigurasi parameter perilaku *user* pada Profile Config

#### 4.2.3. Konfigurasi Pada Simulasi

Setelah mengimplementasikan layanan VoIP kedalam simulasi maka langkah berikutnya yaitu mengimplementasikan parameter jaringan WiMAX ke dalam model jaringan. parameter dalam konfigurasi ini mengacu pada Tabel 4.2 dan parameter-parameter lain yang sesuai dengan standar *mobile* WiMAX 802.16e. Seluruh konfigurasi ini dilakukan di dalam *node model* WiMAX\_Config.

**4.2.3.1. Service Class**

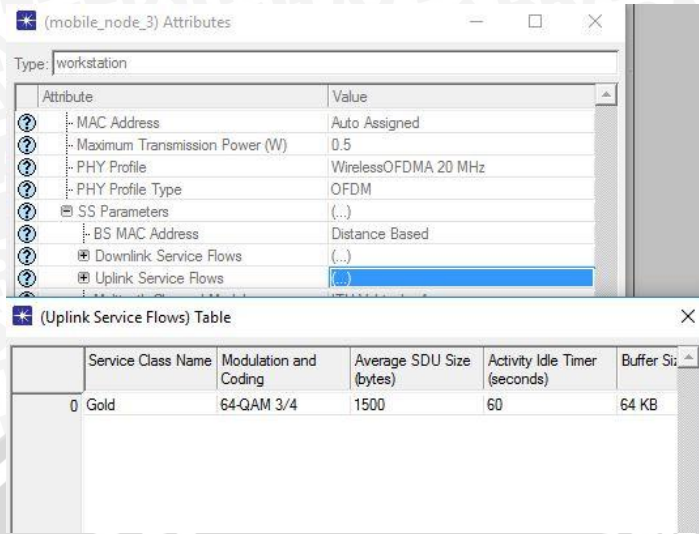
Suatu kelas layanan (*Service Class*) digunakan untuk mengelompokkan kebutuhan QoS sesuai dengan prioritas pengguna. Kelas-kelas layanan tersebut didefinisikan dalam *MAC Service Class Definition*. Secara *default*, terdiri dari tiga kelas layanan, yakni *Gold*, *Silver*, dan *Bronze*. Untuk mengetahui pengaruh *service class* terhadap layanan VoIP maka perlu dilakukan perubahan penggunaan *service class* sehingga dapat dilakukan sesuai Gambar 4.4.



**Gambar 4.4** Tabel *MAC Service Class Definition*

**4.2.3.2. Service Flow**

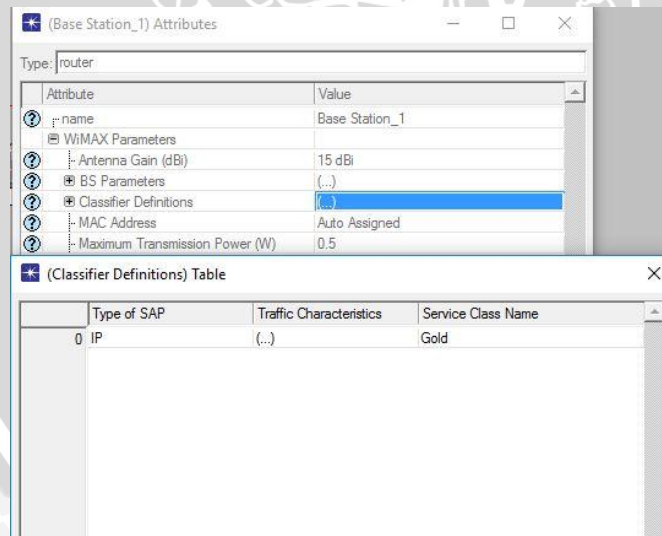
Setelah melakukan konfigurasi terhadap kelas-kelas layanan dalam jaringan WiMAX, kelas-kelas layanan tersebut dapat ditambahkan ke dalam *service flow* yang dibangun antara MS dengan BS. *Uplink* mengacu pada *service flow* dari MS ke BS, sebaliknya *downlink* mengacu pada dari BS ke MS. Pada simulasi ini dilakukan pula perubahan pada *service flow* akan menggunakan kelas layanan *silver*, *gold* dan *bronze* yang mengimplementasikan *service class* seperti yang dapat dilihat pada Gambar 4.5 yang menunjukkan konfigurasi *service flow*.



Gambar 4.5 Konfigurasi Service Flow

**4.2.3.3. Menetapkan Trafik pada Service Classes**

Service flow antara MS dan BS dapat dikonfigurasi dengan menggunakan service class tertentu. Konfigurasi ini dilakukan pada kedua node, baik MS maupun BS. Untuk uplink konfigurasi dilakukan pada sisi MS, sedangkan untuk downlink konfigurasi dilakukan pada sisi BS. Dalam simulasi ini akan digunakan tipe service class Silver. Gambar 4.6 menunjukkan konfigurasi trafik pada service class pada MS dan BS.

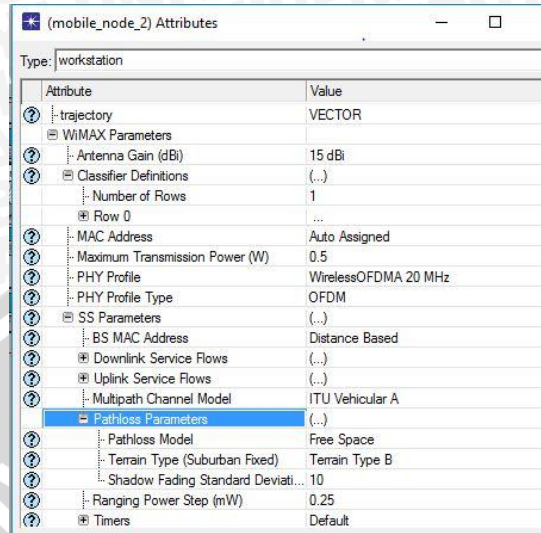


Gambar 4.6 Konfigurasi trafik pada service classes pada BS

**4.2.3.4. Konfigurasi Parameter Physical Layer**

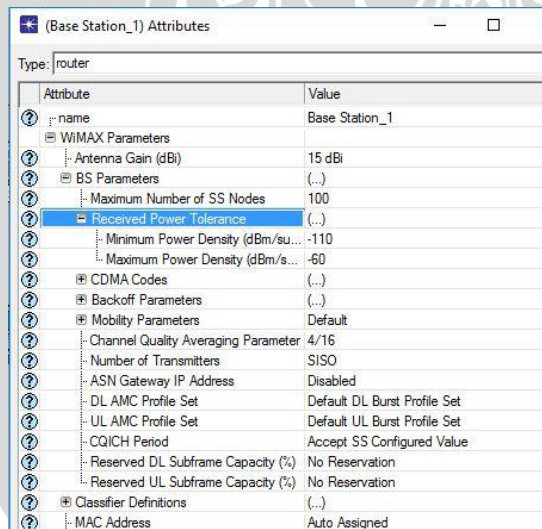
Parameter konfigurasi Physical Layer yang akan dilakukan pada node BS dan MS, tercantum dalam Tabel 4.2. Node model MS digunakan untuk merepresentasikan karakteristik dari

*user yang mobile*. Konfigurasi dilakukan dengan mengedit atribut pada *Wimax\_ss\_wkstn*, sebagaimana ditunjukkan pada Gambar 4.7 berikut ini :



**Gambar 4.7** Konfigurasi pada *node model Mobile Station*

Sedangkan untuk melakukan konfigurasi dalam *node model BS*, dapat dilakukan dengan mengedit atribut pada *Wimax\_bs\_ethernet4\_slip4\_router*, sebagaimana ditunjukkan pada Gambar 4.8 berikut ini :



**Gambar 4.8** Konfigurasi pada *node model Base Station*

#### 4.2.3.5. Konfigurasi Node Model WiMAX

*Node model WiMAX* merupakan model yang digunakan untuk melakukan pengaturan konfigurasi parameter-parameter jaringan WiMAX. Konfigurasi ini mengacu pada parameter yang terdapat pada Tabel 4.3 yang telah disesuaikan dengan standar *mobile WiMAX 802.16e*.

Gambar 4.9 menunjukkan konfigurasi pada *node model WiMAX* :



Attribute	Value
name	node_3
AMC Profile Sets Definitions	(...)
Contention Parameters	(...)
Efficiency Mode	Efficiency Enabled
MAC Service Class Definitions	(...)
Number of Rows	1
Row 0	(...)
OFDM PHY Profiles	(...)
Number of Rows	1
Row 0	(...)
Profile Name	WirelessOFDMA 20 MHz
Frame Duration (milliseconds)	5
Symbol Duration (microseconds)	102.86 (n=28/25, delta_f = 10.94 kHz, ...)
Number of Subcarriers	1024
Frame Structure	(...)
Duplexing Technique	TDD
TC Sublayer Overhead Factor	0
Frequency Band	(...)
Frequency Division	(...)
SC PHY Profiles	(...)

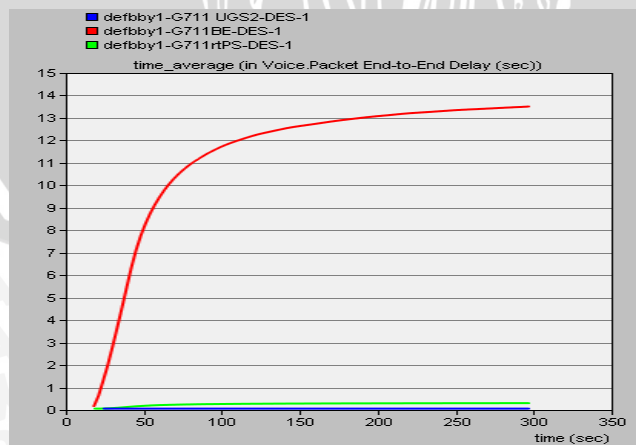
Gambar 4.9 Konfigurasi pada node model WiMAX

## 4.2 Hasil Simulasi dan Pembahasan

Pada sub bab ini membahas hasil penelitian pengaruh variasi *service class* terhadap performansi layanan VoIP dengan menggunakan software OPNET Modeler 14.5 dilihat dari parameter *delay*, *throughput*, dan *packet loss*. Dalam simulasi ini digunakan codec G.711, G.723 dan G.729

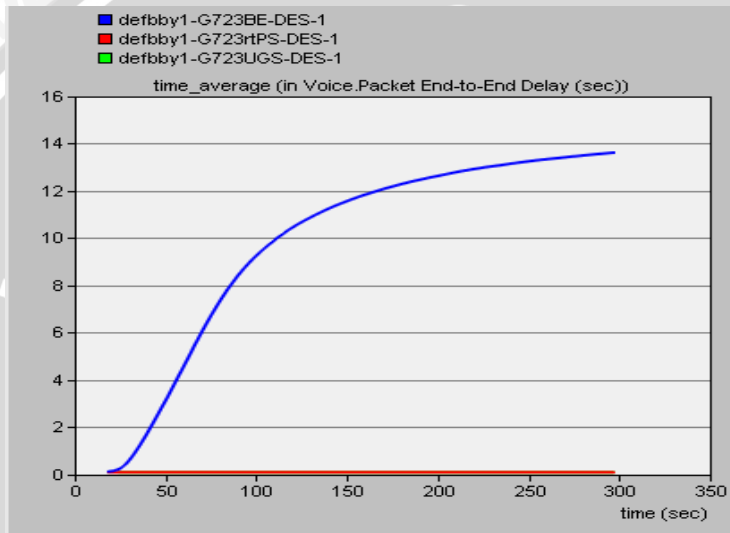
### 4.3.1. Analisis Pengaruh *Service Class* Terhadap *Delay* Layanan VoIP

Pada simulasi ini dilakukan perubahan pada *service class* dan codec layanan VoIP. Hasil simulasi dari pengaruh variasi *service class* terhadap QoS untuk layanan VoIP dengan menggunakan codec G.711 pada *mobile* WiMAX dilihat dari parameter *delay* sehingga didapatkan seperti yang ditunjukkan pada Gambar 4.10.



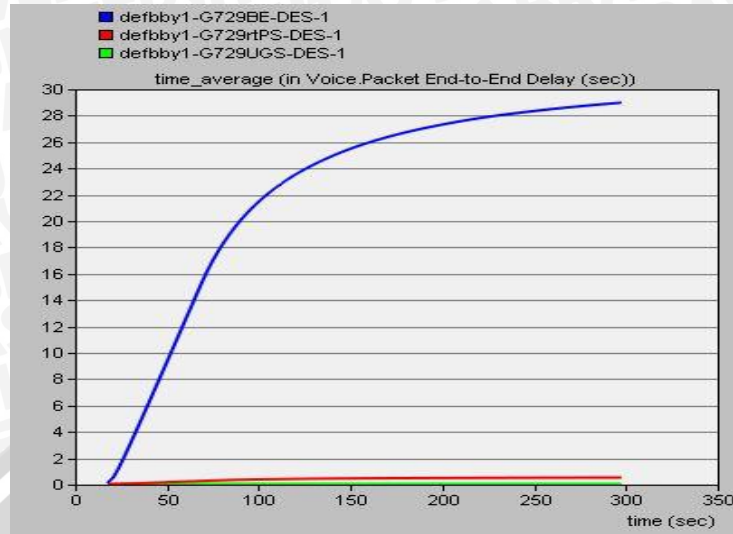
Gambar 4.10 Pengaruh variasi *service class* terhadap *delay* dengan menggunakan codec G.711

Gambar 4.10 menunjukkan grafik dari pengaruh variasi *service class* dengan menggunakan codec G.711 terhadap *delay*. Dapat dilihat bahwa nilai *delay* yang paling besar diberikan saat menggunakan *service class* BE, dibandingkan dengan menggunakan *service class* rtPS dan UGS. Pada UGS *delay* memiliki nilai yang konstan sedangkan pada rtPS *delay* memiliki kenaikan namun tidak besar. Pada simulasi ini digunakan pula codec G.723 sehingga didapatkan hasil simulasi dari pengaruh variasi *service class* terhadap QoS untuk layanan VoIP dengan menggunakan codec G.723 dilihat dari parameter *delay* seperti yang ditunjukkan pada Gambar 4.11.



**Gambar 4.11** Pengaruh variasi *service class* terhadap *delay* dengan menggunakan codec G.723

Gambar 4.11 menunjukkan grafik dari pengaruh variasi *service class* dengan menggunakan codec G.723 terhadap *delay*. Dapat dilihat bahwa dengan menggunakan BE, *delay* mengalami kenaikan yang paling besar. Ketika *service class* UGS *delay* memiliki nilai yang juga konstan dan terlihat sama dengan rtPS. *Delay* rtPS tetap memiliki kenaikan tidak konstan namun karena hanya sedikit sehingga terlihat sama dengan ketika menggunakan *service class* UGS. Ketika digunakan codec G.729 pada simulasi didapatkan hasil dari pengaruh variasi *service class* terhadap QoS untuk layanan VoIP dengan menggunakan codec G.723 dilihat dari parameter *delay* seperti yang ditunjukkan pada Gambar 4.12.



**Gambar 4.12** Pengaruh variasi *service class* terhadap *delay* dengan menggunakan codec G.729

Gambar 4.5 tersebut menunjukkan grafik dari pengaruh variasi *service class* dengan menggunakan codec G.729 terhadap *delay*. Dapat dilihat bahwa dengan menggunakan *service class* BE, *delay* yang paling besar dibandingkan saat menggunakan rtPS dan UGS sedangkan dengan menggunakan rtPS, *delay* memiliki sedikit kenaikan. UGS memiliki *delay* yang konstan.

Selain grafik pada OPNET Modeler 14.5 juga dihasilkan nilai *delay* selama 5 menit layanan VoIP berlangsung dimana setelah dirata-rata maka *delay* yang dihasilkan dapat ditunjukkan pada tabel 4.4. Pada tabel 4.4 diperlihatkan nilai *delay* rata rata dengan variasi *service class* dimana nilai yang paling kecil dihasilkan dari *service class* UGS dengan nilai saat menggunakan codec G.711 adalah 0.060s, codec G.723 adalah 0.100s dan G.729 adalah 0.060s. Sedangkan nilai *delay* rata rata terbesar dihasilkan dari menggunakan *service class* BE dengan nilai saat menggunakan codec G.711 adalah 11.429s, codec G.723 adalah 9.962s dan G.729 adalah 22.222s.

**Tabel 4.4** *delay* rata rata dengan variasi *service class*

Codec	Service class	Rata-rata Delay (s)
G.711	UGS	0.060
	rtPS	0.262
	BE	11.429
G723	UGS	0.100
	rtPS	0.101
	BE	9.962

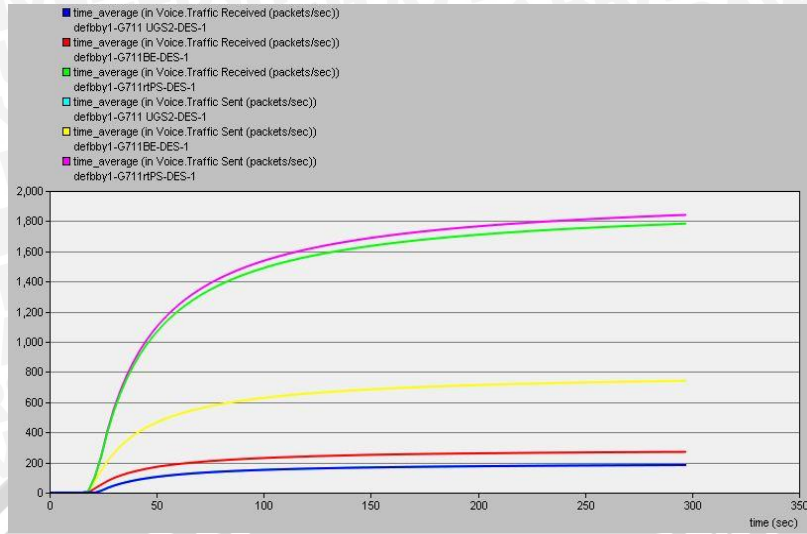
Codec	Service class	Rata-rata Delay (s)
G729	UGS	0.060
	rtPS	0.412
	BE	22.222

Berdasarkan hasil simulasi yang sudah didapatkan maka *delay* yang paling besar diberikan saat BE baik menggunakan codec G7.11, G.723 dan G.729. Hal ini menunjukkan bahwa BE memiliki waktu total yang dibutuhkan untuk mengirimkan paket data dari sumber sampai ke tujuan yang paling besar dibandingkan dengan UGS dan rtPS. Hal ini dikarenakan BE didesain untuk aplikasi yang tidak perlu jaminan *delay* dimana dalam pentransmisian waktu antar *packet* variable. Sedangkan *delay* yang paling kecil diberikan oleh UGS baik dengan menggunakan codec G.711, G.723 dan G.729. Hal ini dikarenakan UGS memiliki waktu pengiriman yang periodik dan konstan sehingga setiap pengiriman data menggunakan UGS maka akan memiliki *delay* yang konstan dan kecil. Sehingga dapat diketahui bahwa codec dan *service class* mempengaruhi nilai dari *delay* dari layanan VoIP.

Namun pada codec G.729 BE memberikan *delay* yang paling besar diantara saat codec yang lain. Hal ini dikarenakan codec G.729 memiliki *delay* codec yang paling besar yaitu 97,5 ms, dimana *delay end to end* merupakan hasil penjumlahan dari *delay* codec dan delay MAN. Sedangkan delay yang paling kecil diberikan pada codec G.723 dimana

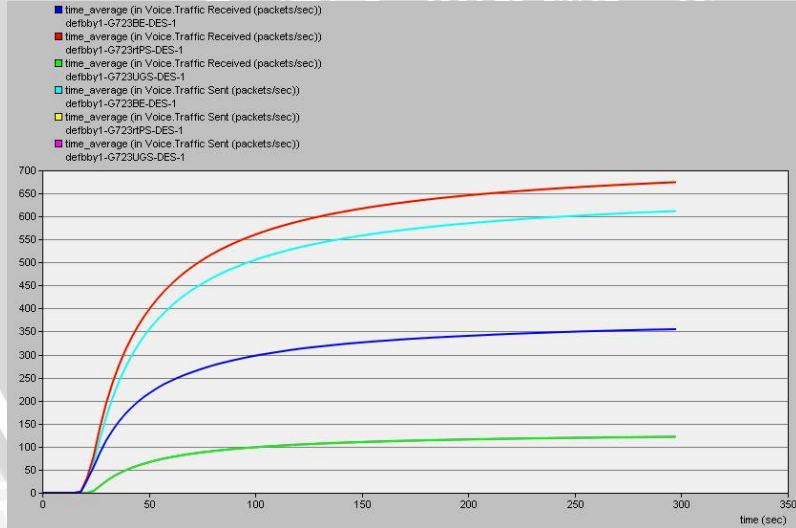
#### 4.3.2. Analisis Pengaruh Service Class Terhadap Packet Loss Layanan VoIP

Hasil simulasi untuk pengaruh variasi *service class* terhadap QoS untuk layanan VoIP pada *mobile* WiMAX dilihat dari parameter *delay* yang didapatkan seperti yang ditunjukkan pada Gambar 4.13. Gambar tersebut menunjukkan grafik *traffic sent* dan *traffic received* dimana *packet loss* dihasilkan dari pengurangan *traffic sent* dengan *traffic received*. Dapat dilihat bahwa dengan menggunakan *service class* BE memberikan nilai *packet loss* yang paling besar dibandingkan dengan UGS dan rtPS.



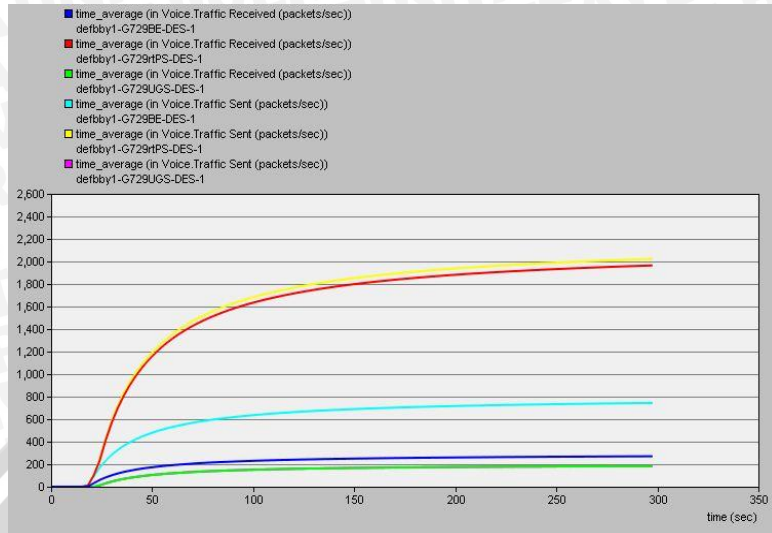
Gambar 4.13 Pengaruh variasi *service class* terhadap *packet loss* dengan menggunakan codec G.711

Pada Gambar 4.14 menunjukkan grafik dari pengaruh variasi *service class* dengan menggunakan codec G.723 terhadap *packet loss*. Dapat dilihat bahwa dengan menggunakan *service class* BE memberikan nilai yang paling besar dibandingkan dengan UGS dan rtPS. Sedangkan UGS memiliki nilai *packet loss* yang paling kecil.



Gambar 4.14 Pengaruh variasi *service class* terhadap *packet loss* dengan menggunakan codec G.723

Pada Gambar 4.15 menunjukkan grafik dari pengaruh variasi *service class* dengan menggunakan codec G.711 terhadap *packet loss*. Dapat dilihat bahwa dengan menggunakan *service class* BE memberikan nilai yang paling besar. Sedangkan UGS memiliki nilai *packet loss* yang paling kecil.



**Gambar 4.15** Pengaruh variasi *service class* terhadap *packet loss* dengan menggunakan codec G.729

Dari hasil simulasi didapatkan nilai rata rata *traffic received* dan *traffic sent* seperti yang ditunjukkan pada tabel 4.5.

**Tabel 4.5** nilai rata rata *traffic received* dan *traffic sent*

Codec	Service Class	Traffic Received (packets/s)	Traffic Sent (packets/s)
G 7.11	UGS	142.4781	142.4791
	rtPS	1404.2200	1449.4660
	BE	216.1637	591.7321
G 7.23	UGS	93.5462	93.3546
	rtPS	528.8739	528.9139
	BE	280.8196	477.6423
G 7.29	UGS	142.4617	142.4627
	rtPS	1541.9140	1587.5530
	BE	216.6414	597.7833

Packet loss dihasilkan dari *traffic received* (packet/second) dan *traffic sent* (packet/second) dimana dengan menggunakan rumus

$$\rho_{total} = \frac{traffic\ sent - traffic\ received}{traffic\ sent}$$

dengan :

$\rho_{tot}$  : Packet loss total

Maka nilai packet loss dapat dihasilkan dari

$$\rho_{total} = \frac{142.4791 - 142.4781}{142.4791} = 0.000007 \text{ packets/s}$$

Sehingga packet loss dengan variasi *service class* ditunjukkan pada tabel 4.6 dimana nilai yang paling kecil dihasilkan dari *service class* UGS baik menggunakan codec G 7.11, G 7.23 dan G.729. Sedangkan *packet loss* terbesar dihasilkan dari menggunakan *service class* BE baik dengan menggunakan codec G.711, G.723 dan G.729.

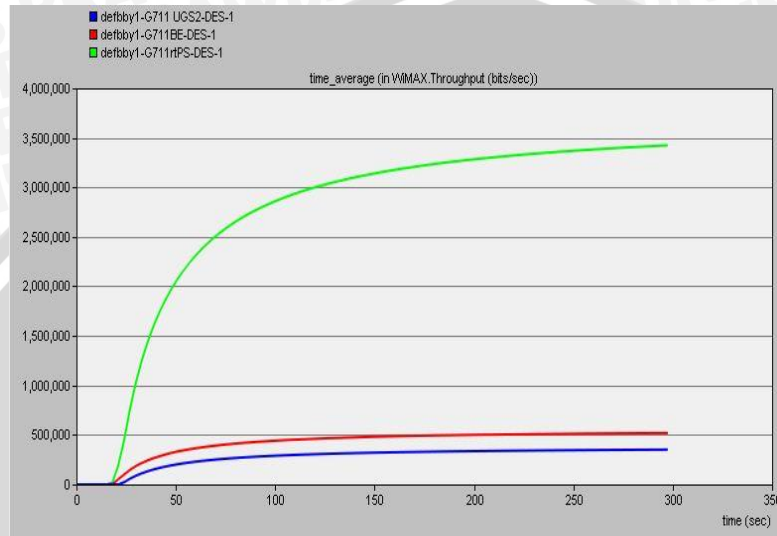
**Tabel 4.6** *packet loss* dengan variasi *service class*

Codec	Service Class	Packet Loss (pakket/s)
G.711	UGS	0.000007
	rtPS	0.031215
	BE	0.634693
G.723	UGS	0.000001
	rtPS	0.000756
	BE	0.412071
G.729	UGS	0.000007
	rtPS	0.028748
	BE	0.637592

Berdasarkan hasil simulasi yang sudah didapatkan maka *packet loss* yang paling besar diberikan saat BE baik menggunakan codec G7.11, G.723 dan G.729. Hal ini menunjukkan bahwa BE memiliki paket data yang paling banyak salahnya saat diterima oleh penerima, dikarenakan BE didesain dengan pengiriman data berupa paket data yang berukuran variable dengan waktu yang variabel pula sehingga dapat memberikan error yang lebih banyak dibandingkan dengan UGS dan rtPS. Sedangkan *packet loss* yang paling kecil diberikan oleh UGS baik dengan menggunakan codec G7.11, G.723 dan G.729. Hal ini dikarenakan UGS dirancang untuk mendukung aliran data *real-time* yang terdiri dari paket data dengan ukuran yang tetap yang dikeluarkan pada interval periodic. Packet loss dipengaruhi oleh mekanisme bentuk paket yang dikirimkan oleh tiap *service class*.

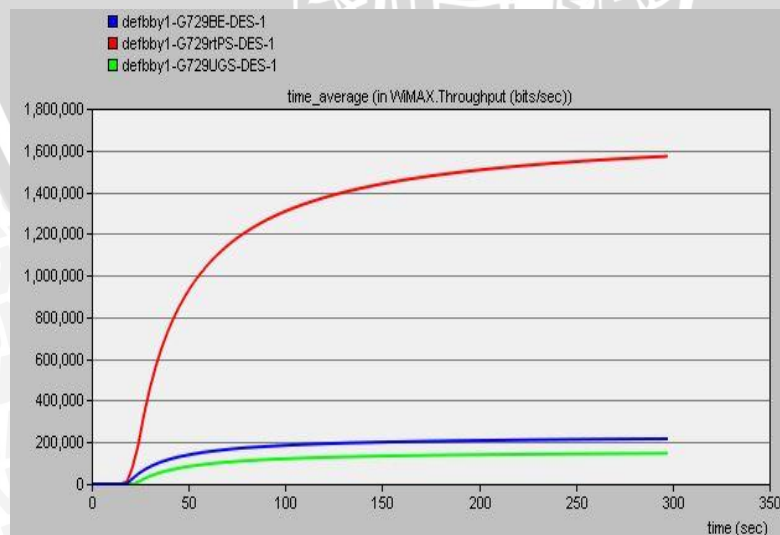
### 4.3.3. Analisis Pengaruh *Service Class* Terhadap *Throughput* Layanan VoIP

Hasil simulasi untuk *throughput* dengan menggunakan variasi *service class* seperti yang ditunjukkan pada Gambar 4.16 dimana dengan rtPS *throughput* lebih besar dibandingkan dengan UGS dan BE. Sedangkan yang paling kecil dihasilkan oleh UGS.



Gambar 4.16 Pengaruh *service class* terhadap *throughput* dengan menggunakan codec G.711

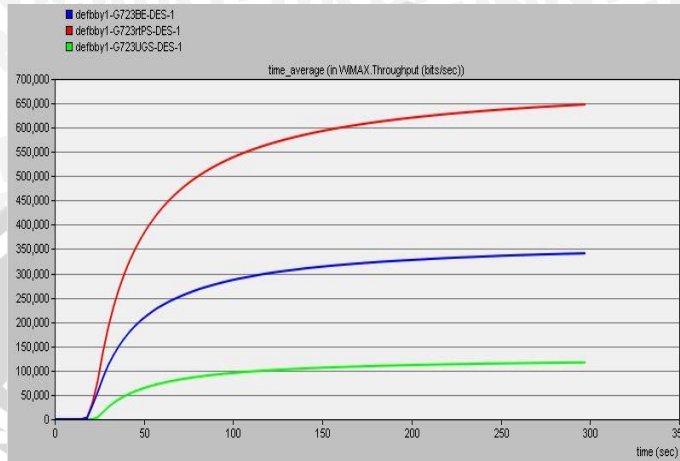
Pada Gambar 4.17 ditunjukkan nilai *throughput* variasi *service class* dengan menggunakan codec G.729 dimana dengan rtPS *throughput* lebih besar dibandingkan dengan *service class* BE dan UGS. BE dan UGS memiliki *throughput* yang hampir sama sedangkan yang paling kecil dihasilkan oleh UGS.



Gambar 4.17 Pengaruh *service class* terhadap *throughput* dengan menggunakan codec G.729



Pada Gambar 4.18 pengaruh *service class* terhadap *throughput* dengan menggunakan codec G.723 dimana dengan rtPS *throughput* lebih besar dibandingkan dengan *service class* yang lain. Sedangkan yang paling kecil dihasilkan oleh UGS. Nilai *throughput* BE dan UGS memiliki perbedaan yang cukup jauh.



**Gambar 4.18** Pengaruh *service class* terhadap *throughput* dengan menggunakan codec G.723

Pada tabel 4.7 ditunjukkan nilai *throughput* dengan variasi *service class* dimana nilai yang paling kecil dihasilkan dari *service class* UGS baik saat menggunakan codec G.711, G.723 dan G.729. Sedangkan *throughput* terbesar dihasilkan dari menggunakan *service class* rtPS baik dengan menggunakan codec G.711, G.723 dan G.729.

**Tabel 4.7** *Throughput* dengan variasi *service class*

Codec	Service Class	Throughput (bit/s)
G.711	UGS	273665
	rtPS	2696785
	BE	415703
G.723	UGS	89952
	rtPS	508396
	BE	270290
G.729	UGS	114075
	rtPS	1234170
	BE	173989

Berdasarkan hasil simulasi yang sudah didapatkan maka *throughput* yang paling besar diberikan saat rtPS baik menggunakan codec G7.11, G.723 dan G.729. Hal ini

menunjukkan bahwa rtPS memiliki ukuran yang menyatakan berapa banyak bit yang dapat ditransmisikan dan sukses diterima di tujuan per detik untuk lebar pita yang dialokasikan dikarenakan rtPS didesain untuk aplikasi yang sensitif terhadap *throughput*. Sedangkan *packet loss* yang paling kecil diberikan oleh UGS baik dengan menggunakan codec G7.11, G.723 dan G.729.

