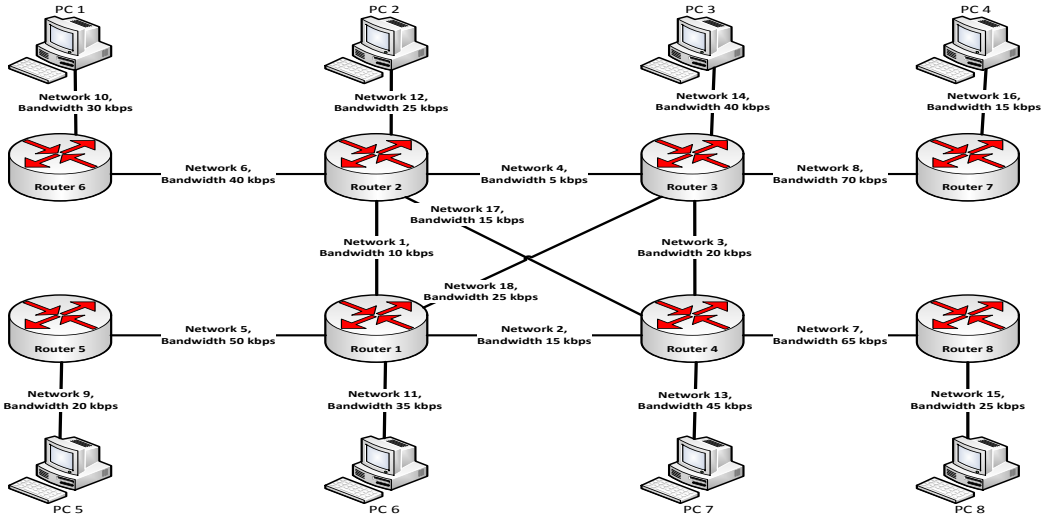
BAB IV
IMPLEMENTASI DAN PENGUJIAN OPTIMASI ROUTING OSPF MENGGUNAKAN MODEL MPLS TRAFFIC ENGINEERING

Tahap implementasi merupakan tahapan penerapan dari parameter serta komponen yang sudah dilakukan pada tahapan analisis pada teknologi maupun model sebelumnya dengan model yang sudah dipilih dan diseleksi untuk dilakukan penilaian pada tahap pengujian.

IV.1. Implementasi Skema Jaringan

Spesifikasi skema jaringan yang digunakan untuk implementasi mencakup desain topologi yang menunjukkan konektivitas fisik komputer (*client* maupun *server*) pada suatu jaringan komputer serta segmentasi alamat yang digunakan pada jaringan tersebut. Skema jaringan yang digunakan untuk implementasi adalah contoh model skema jaringan komputer yang diadopsi dari penerapan jaringan *inherent* sekolah SMKN 1 Cimahi menjadi sebuah model baru yang akan disimulasikan pada simulator jaringan *omnet++*. Skema jaringan yang akan diadopsi tersebut seperti yang terlihat pada Gambar IV-1.



Gambar IV-1 Skema Jaringan Komputer

Sedangkan segmentasi dari skema jaringan komputer yang digunakan pada Gambar IV-1 disajikan pada Tabel IV-1 di bawah ini.

Tabel IV-1 Skema Segmentasi Jaringan

Network	IP Address	*Bandwidth*
Network 1	10.10.1.0/24	*Bandwidth* 60 kbps
Network 2	10.10.2.0/24	*Bandwidth* 80 kbps
Network 3	10.10.3.0/24	*Bandwidth* 70 kbps
Network 4	10.10.4.0/24	*Bandwidth* 40 kbps
Network 5	10.10.5.0/24	*Bandwidth* 50 kbps
Network 6	10.10.6.0/24	*Bandwidth* 40 kbps
Network 7	10.10.7.0/24	*Bandwidth* 65 kbps
Network 8	10.10.8.0/24	*Bandwidth* 70 kbps
Network 9	10.10.9.0/24	*Bandwidth* 20 kbps
Network 10	10.10.10.0/24	*Bandwidth* 30 kbps
Network 11	10.10.11.0/24	*Bandwidth* 35 kbps
Network 12	10.10.12.0/24	*Bandwidth* 25 kbps
Network 13	10.10.13.0/24	*Bandwidth* 45 kbps
Network 14	10.10.14.0/24	*Bandwidth* 40 kbps
Network 15	10.10.15.0/24	*Bandwidth* 25 kbps
Network 16	10.10.16.0/24	*Bandwidth* 15 kbps
Network 17	10.10.17.0/24	*Bandwidth* 50 kbps
Network 18	10.10.18.0/24	*Bandwidth* 85 kbps

IV.1.1. Komponen Implementasi

Komponen-komponen serta parameter-parameter yang dibutuhkan terkait untuk mendukung pada tahapan implementasi diantaranya:

1. Perangkat Keras

   Perangkat keras yang dimaksud dalam kebutuhan perangkat keras pada tahap implementasi ini adalah seperangkat komputer maupun laptop yang di *install* aplikasi simulator jaringan OMNET++ yang akan digunakan untuk melakukan implementasi model jaringan yang sudah dipilih untuk di lakukan suatu simulasi. Perangkat keras yang dibutuhkan dapat dilihat pada Tabel IV-2 di bawah ini.

Tabel IV-2 Kebutuhan Perangkat Keras

Perangkat Keras	Spesifikasi
Processor	Minimum QuadCore 2.8 Ghz
Harddisk	Minimum 40 GB

Perangkat Keras	Spesifikasi
Monitor	CRT/LCD minimum resolusi 1024x768
RAM	Minimum 4 GB
Mouse + Keyboard	-

2. Perangkat Lunak

   Perangkat lunak yang dimaksud dalam kebutuhan perangkat lunak pada tahap implementasi ini adalah perangkat lunak yang dibutuhkan untuk dapat dilakukan instalasi perangkat lunak simulator jaringan OMNET++. Perangkat lunak yang dibutuhkan dapat dilihat pada tabel spesifikasi *platform* sistem operasi windows pada Tabel IV-3.

Tabel IV-3 Spesifikasi Platform Windows

Perangkat Lunak	Spesifikasi
Sistem Operasi Windows	Minimum XP, 7, dan 8 (32 maupun 64 bit)
C++ Compiler	Minimum gcc versi 4.x, mingwenv-tools

Kebutuhan perangkat lunak untuk *platform* linux dapat dilihat pada tabel spesifikasi *platform* sistem operasi linux pada Tabel IV-4.

Tabel IV-4 Spesifikasi Platform Linux

Perangkat Lunak	Spesifikasi
Sistem Operasi Linux	Minimum ubuntu 12.04 LTS, Fedora 18, Redhat 6.4, dan OpenSuse 12.3
C++ Compiler	Build-essential, gcc, g++, bison, flex, perl, tcl-dev, tk-dev, blt, libxml2-dev zlib1g-dev, openjdk-6-jre, doxygen, graphviz, openmpi-bin, libopenmpi-dev, libpcap-dev

3. Simulator Jaringan OMNET++

   Simulator jaringan OMNET++ yang digunakan pada implementasi ini menggunakan OMNET++ versi 4.4.1 untuk *platform* windows 32 bit, kemudian *library* model simulasi jaringan yang digunakan adalah INET *framework* versi 2.2 yang didalamnya mencakup kebutuhan untuk memodelkan dan mensimulasikan konsep TCP/IP, IPv4, IPv6, ethernet, PPP, protokol 802.11,OSPF dan MPLS.

Komponen lainnya yang terkait implementasi untuk melakukan simulasi jaringan pada simulator jaringan OMNET++ adalah komponen yang dapat mendukung untuk memodelkan skema jaringan yang sudah ditentukan sebelumnya seperti:

1. Router Model

   Router yang diimplementasikan pada simulator jaringan OMNET++ digunakan untuk memodelkan dan mensimulasikan suatu node dalam jaringan komputer yang bertugas menentukan dan mencari rute jalur untuk suatu paket data saat komunikasi data berlangsung, meneruskan paket data yang dikirimkan dari komputer pengirim menuju komputer tujuan penerima paket data, sampai pada melakukan simulasi *load balancing* beban data pada suatu jalur komunikasi data.

2. Switch Model

   Switch yang diimplementasikan pada simulator jaringan OMNET++ digunakan untuk memodelkan dan mensimulasikan suatu terminal dalam jaringan komputer yang bertugas sebagai titik percabangan dari proses transfer data dalam satu jaringan.

3. Komputer Model

   Komputer model yang diimplementasikan pada OMNET++ digunakan untuk memodelkan dan mensimulasikan suatu node dalam jaringan komputer yang bertugas sebagai pengirim maupun penerima suatu paket data dalam proses komunikasi data. Komputer model pada implementasi dimodelkan dengan model komputer *client* dan model komputer *server*.

4. Konektifitas model

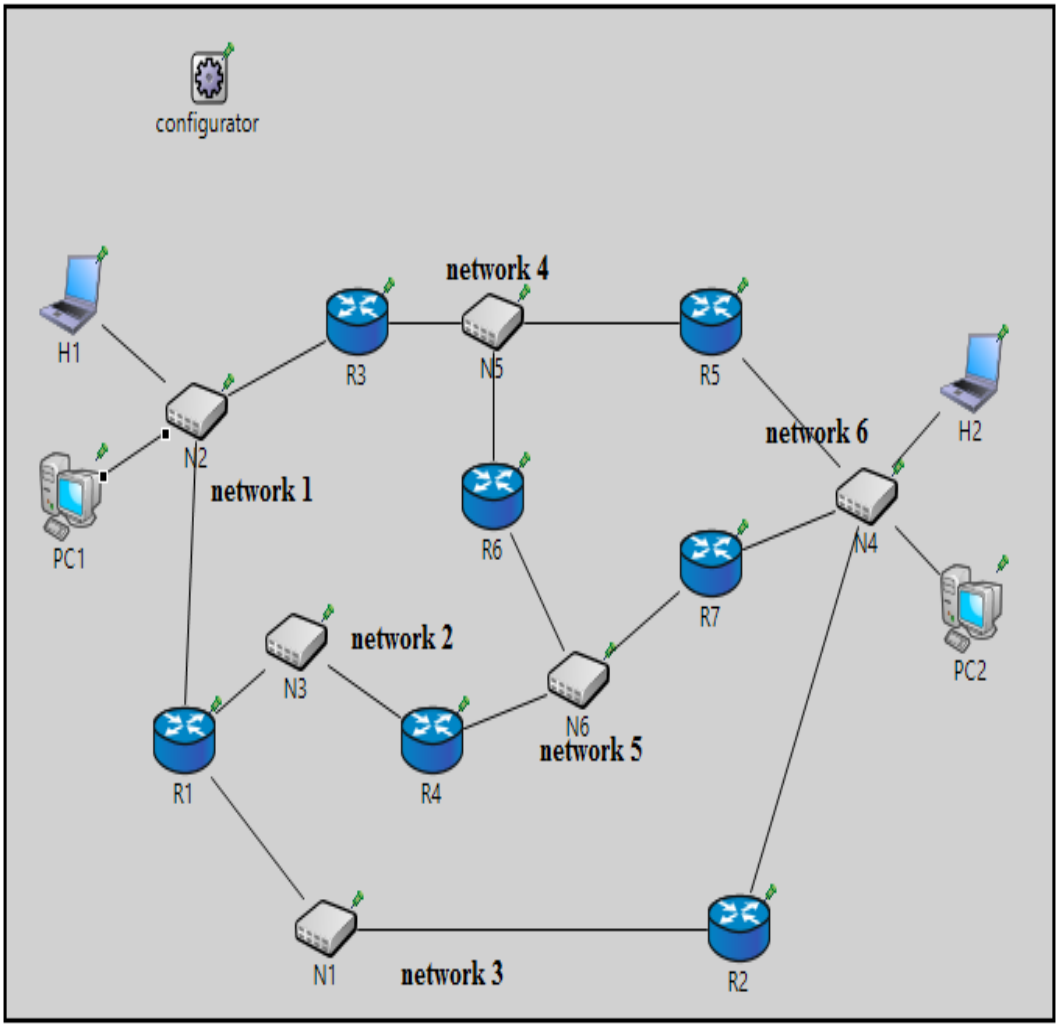
   Konektifitas model yang diimplementasikan pada OMNET++ digunakan untuk memodelkan dan mensimulasikan suatu media penghubung jaringan yang membentuk suatu topologi jaringan yang saling memiliki keterhubungan satu sama lain pada skema jaringan yang digunakan pada model jaringan. Konektifitas model yang digunakan pada implementasi digunakan model konektifitas media kabel (*wireline*).

IV.1.2. Implementasi Topologi

Implementasi topologi yang akan digunakan pada pembahasan ini akan menggunakan beberapa model topologi yang diadopsi dari contoh skema jaringan seperti yang terlihat pada Gambar IV-1 di pembahasan sebelumnya. Topologi yang diadopsi dari contoh skema jaringan tersebut selanjutnya akan dimodelkan pada *tools* simulator jaringan OMNET++ dan dibentuk menjadi dua model. Masing-masing model topologi jaringan tersebut diantaranya:

1. Model Topologi non Optimasi

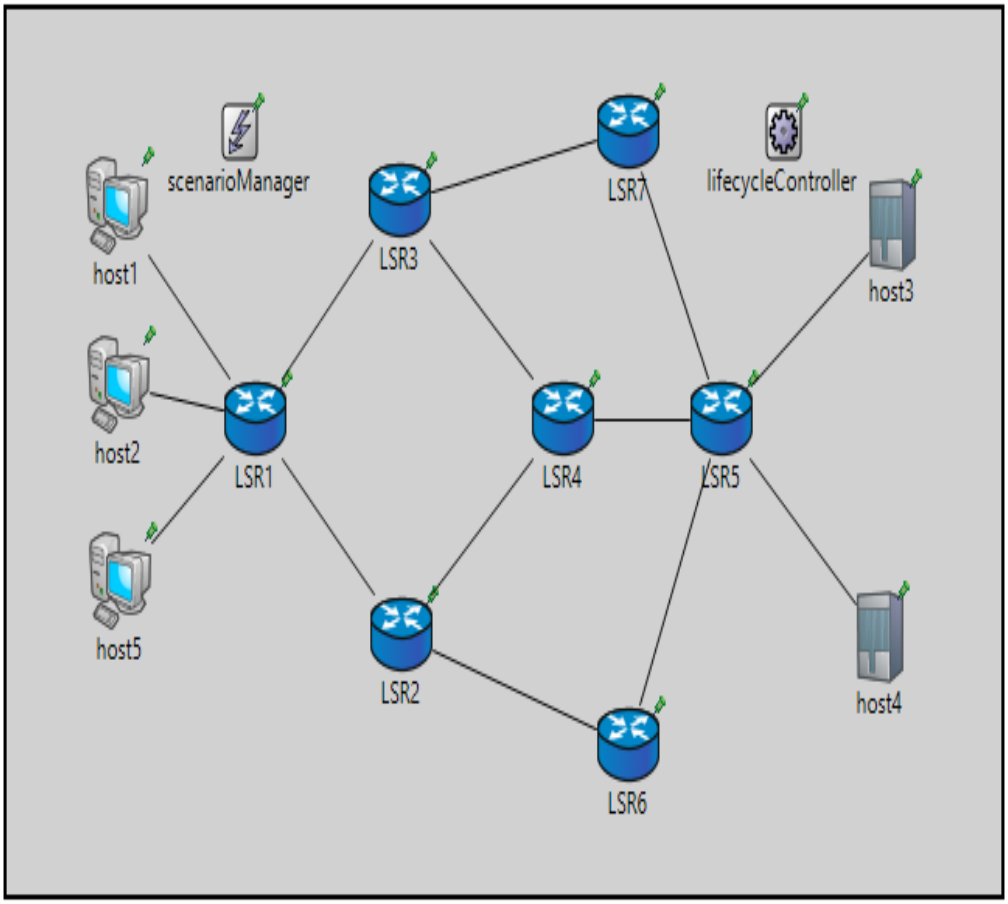
   Topologi non optimasi akan digunakan untuk melakukan simulasi dan pengujian pada model jaringan dengan protokol routing OSPF seperti yang terlihat pada Gambar IV-2.



Gambar IV-2 Model Topologi non Optimasi

2. Model Topologi Optimasi

   Topologi Optimasi akan digunakan untuk melakukan simulasi dan pengujian pada model jaringan dengan protokol routing OSPF menggunakan model multi protokol label switching seperti yang terlihat pada Gambar IV-3.



Gambar IV-3 Model Topologi Optimasi

IV.2. Pengujian

Pengujian merupakan rangkaian tahapan yang dilakukan untuk menguji serta mengetahui nilai dari parameter-parameter yang diujikan dari suatu bentuk model yang sudah disimulasikan dengan tujuan untuk mengetahui kekurangan maupun kelebihan dari model topologi yang telah disimulasikan. Pengujian yang akan dilakukan mencakup pengujian terhadap parameter-parameter dan komponen-komponen yang terkait dengan optimasi protokol *routing* pada jaringan komputer yang akan disimulasikan pada model non optimasi dan model

optimasi untuk diuji sesuai dengan beberapa rencana dan skenario pengujian yang sudah disiapkan.

IV.2.1. Rencana Pengujian

Rencana pengujian akan dilakukan dengan cara mensimulasikan kedua model jaringan yang telah dijelaskan pada sub bab point IV.1.2 tentang implementasi topologi diantaranya adalah topologi non optimasi dan topologi optimasi. Model topologi akan disimulasikan pada *tools* simulator jaringan OMNET++ dan *library* INET *framework* versi 2.2 yang didalamnya mencakup kebutuhan untuk memodelkan dan mensimulasikan konsep TCP/IP, IPv4, IPv6, ethernet, PPP, protokol 802.11,OSPF dan MPLS. Parameter-parameter pengujian fokus pada permasalahan optimasi, pengujian dilakukan terhadap parameter-parameter yang terkait optimasi (*quality of service*) pada jaringan komputer seperti *throughput, packet delay, packet delivery ratio,* dan *routing information.*

IV.2.2. Skenario Pengujian

Skenario pengujian disini digunakan sebagai acuan pada saat melakukan proses pengujian dan simulasi, adapun skenario pengujian yang akan dilakukan dibuat menjadi beberapa skenario uji yang terkait dengan optimasi protokol *routing* OSPF pada jaringan komputer. Skenario pengujian yang akan dilakukan fokus pada permasalahan optimasi, pengujian dilakukan terhadap parameter-parameter yang terkait optimasi (*quality of service*) pada jaringan komputer seperti *throughput, packet delay, packet delivery ratio,* dan *routing information.*

Berdasarkan parameter-parameter tersebut, maka akan dibuat tiga skenario utama pengujian yang akan disimulasikan baik pada model jaringan non optimasi maupun model jaringan optimasi. Skenario-skenario utama untuk pengujian tersebut diantaranya:

1. Skenario model simulasi transmisi data *reply access* menggunakan protokol ICMP dilakukan secara normal sesuai dengan data yang dikirimkan dari pengirim menuju penerima pada jalur jaringan multi jalur.

2. Skenario model simulasi jalur yang digunakan ketika proses komunikasi data berlangsung sesuai dengan jalur yang sudah ditetapkan oleh protokol *routing* pada tabel *routing information* pada jalur jaringan multi jalur.
3. Skenario model simulasi dengan menggunakan suatu beban data berupa *voice message* yang dikirimkan dari pengirim menuju penerima pada jalur jaringan multi jalur.

Alasan menggunakan skenario tersebut karena untuk dapat mengetahui dan melihat terkait optimasi (*quality of service*) pada jaringan komputer seperti *throughput, packet delay, packet delivery ratio,* dan *routing information* serta bagaimana proses aliran data pada suatu jaringan komputer multi jalur yang dikirimkan dari pengirim menuju penerima sebagai mekanisme transmisi data saat terjadi proses komunikasi data dialirkan pada jalur jaringan sesuai dengan *routing information* pada tabel *routing* dengan mekanisme jalur yang sudah ditetapkan oleh suatu protokol *routing*. Parameter-parameter terkait optimasi (*quality of service*) pada proses transmisi data dari pengirim yang dilakukan secara *unicast* akan dibandingkan dan dievaluasi bagaimana pengaruhnya dengan mekanisme sebelum dan sesudah diterapkannya teknologi *traffic engineering* MPLS pada jaringan komputer multi jalur.

Setiap skenario yang dievaluasi dari model simulasi akan dijalankan oleh *tools* simulator jaringan komputer OMNET++ dan hasil simulasi yang terekam akan ditampilkan berupa data vector, skalar dan berupa data yang ditampilkan dalam *sequential chart information.* Berikut ini adalah beberapa spesifikasi parameter yang akan diujikan pada skenario utama pada setiap model topologi dalam mensimulasikan skenario tersebut, baik pada model topologi non optimasi maupun pada topologi optimasi yang disimulasikan:

1. Skenario evaluasi proses transmisi data *reply access* menuju *network* selanjutnya akan diulangi sebanyak 15 kali untuk setiap pesan data yang dikirimkan.

2. Skenario evaluasi proses untuk interval waktu pengiriman paket data berupa ICMP *message* setiap 0.1 detik.
3. Skenario evaluasi proses untuk ukuran paket data *voice* yang disimulasikan sebesar 64 *bytes*, sedangkan untuk ukuran paket data ICMP *message* yang disimulasikan sebesar 32 *bytes*.
4. Skenario evaluasi untuk alokasi alamat IP menggunakan alamat IPv4 dan *routing information table* dari setiap node *router* di set secara otomatis oleh modul *network configurator4* pada setiap simulasi dijalankan.

IV.2.3. Hasil Pengujian

Berdasarkan pada parameter-parameter pengujian yang telah didefinisikan, maka parameter yang digunakan untuk melakukan penilaian dan evaluasi mekanisme sebelum dan sesudah diterapkannya teknologi *traffic engineering* MPLS pada jaringan komputer multi jalur seperti *throughput*, *packet delay*, *packet delivery ratio,* dan *routing information.*

Pengujian pertama akan dilakukan berupa simulasi pada model topologi non optimasi yang menerapkan protokol *routing* OSPF (*Open Shortest Path First*), *source code* yang diterapkan di setiap node (*router*) sebagai berikut:

```
<?xml version="1.0"?>
<OSPFASConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="OSPF.xsd">

  <!-- Areas -->
  <Area id="0.0.0.0">
    <AddressRange address="R1>N1" mask="R1>N1/" status="Advertise"
/>
    <AddressRange address="R1>N2" mask="R1>N2/" status="Advertise"
/>
    <AddressRange address="R1>N3" mask="R1>N3/" status="Advertise"
/>
    <AddressRange address="R2>N4" mask="R2>N4/" status="Advertise"
/>
    <AddressRange address="R3>N5" mask="R3>N5/" status="Advertise"
/>
    <AddressRange address="R4>N6" mask="R4>N6/" status="Advertise"
/>
  </Area>

  <!-- Routers -->
  <Router name="R1" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
```

```
interfaceOutputCost="1" routerPriority="1" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="1" routerPriority="1" />
    <BroadcastInterface ifName="eth2" areaID="0.0.0.0"
interfaceOutputCost="1" routerPriority="1" />
  </Router>

  <Router name="R2" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="2" routerPriority="2" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="2" routerPriority="2" />
  </Router>

  <Router name="R3" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="3" routerPriority="3" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="3" routerPriority="3" />
  </Router>

  <Router name="R4" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="4" routerPriority="4" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="4" routerPriority="4" />
  </Router>

  <Router name="R5" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="5" routerPriority="5" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="5" routerPriority="5" />
  </Router>

  <Router name="R6" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="6" routerPriority="6" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="6" routerPriority="6" />
  </Router>

  <Router name="R7" RFC1583Compatible="true">
    <BroadcastInterface ifName="eth0" areaID="0.0.0.0"
interfaceOutputCost="7" routerPriority="7" />
    <BroadcastInterface ifName="eth1" areaID="0.0.0.0"
interfaceOutputCost="7" routerPriority="7" />
  </Router>

</OSPFASConfig>
```

Skenario utama yang digunakan untuk melakukan pengujian menggunakan skenario pengujian yang sudah dijelaskan sebelumnya untuk menilai *throughput*

saat komunikasi data berlangsung. *Source code* yang diterapkan untuk memodelkan dan mensimulasikan topologi non optimasi sebagai berikut:

```
[General]
description = "Backbone test"
network = Backbone
tkenv-plugin-path = ../../../etc/plugins

**.ospf.ospfConfig = xmldoc("ASConfig.xml")

**.numUdpApps = 3
**.udpApp[0].typename = "UDPBasicApp"
**.udpApp[0].destPort = 1234
**.udpApp[0].messageLength = 32 bytes
**.udpApp[0].sendInterval = 0.1s
**.H2.udpApp[0].destAddresses = "192.168.2.2"
**.H1.udpApp[0].destAddresses = "192.168.4.1"

**.udpApp[1].typename = "UDPVideoStreamSvr"
**.udpApp[1].destPort = 3088
**.udpApp[1].messageLength = 64 bytes
**.udpApp[1].sendInterval = 0.1s
**.PC2.udpApp[1].destAddresses = "192.168.2.4"
**.PC1.udpApp[1].destAddresses = "192.168.4.5"

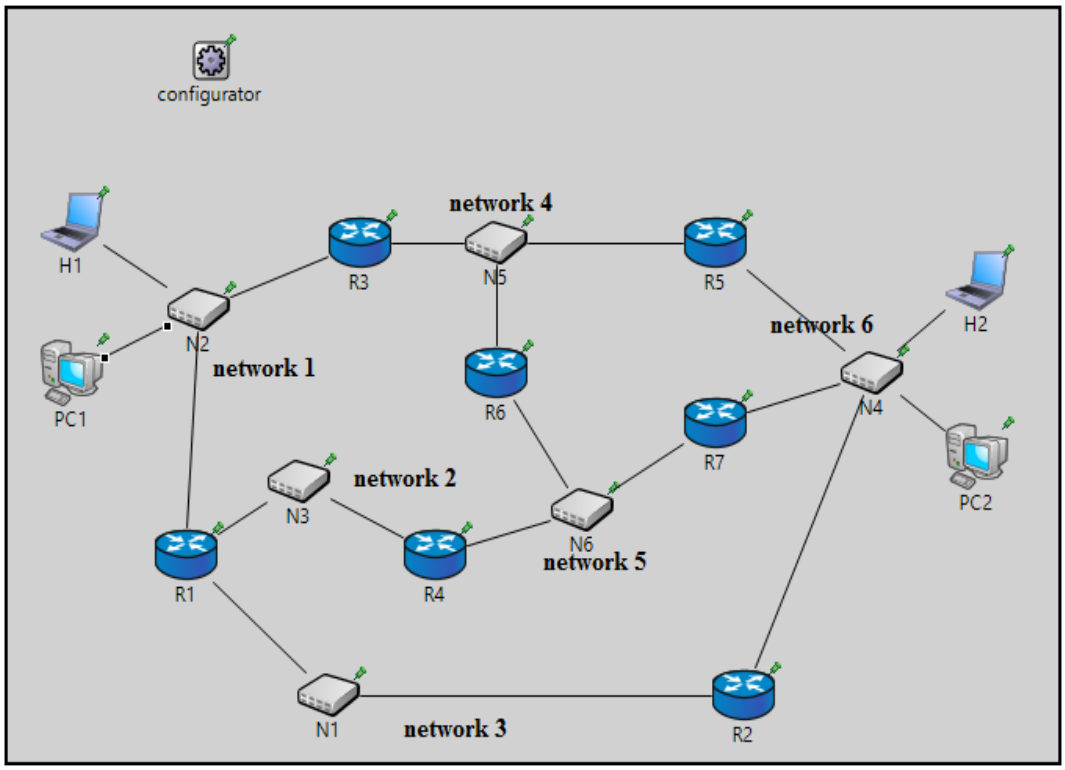
**.udpApp[2].typename = "UDPEchoApp"
**.udpApp[2].localPort = 1234

**.lossRate.result-recording-modes = all
**.channel.throughput.result-recording-modes = all

**.eth[*].queueType = "DropTailQueue"
**.eth[*].queue.*.frameCapacity = 150

**.R1.routingTable.module-eventlog-recording = true
**.R2.routingTable.module-eventlog-recording = true
**.R3.routingTable.module-eventlog-recording = true
**.R4.routingTable.module-eventlog-recording = true
**.R5.routingTable.module-eventlog-recording = true
**.R6.routingTable.module-eventlog-recording = true
**.R7.routingTable.module-eventlog-recording = true
**.arp.cacheTimeout = 1s
```

Rancangan model topologi non optimasi yang digunakan seperti yang terlihat pada Gambar IV-4.



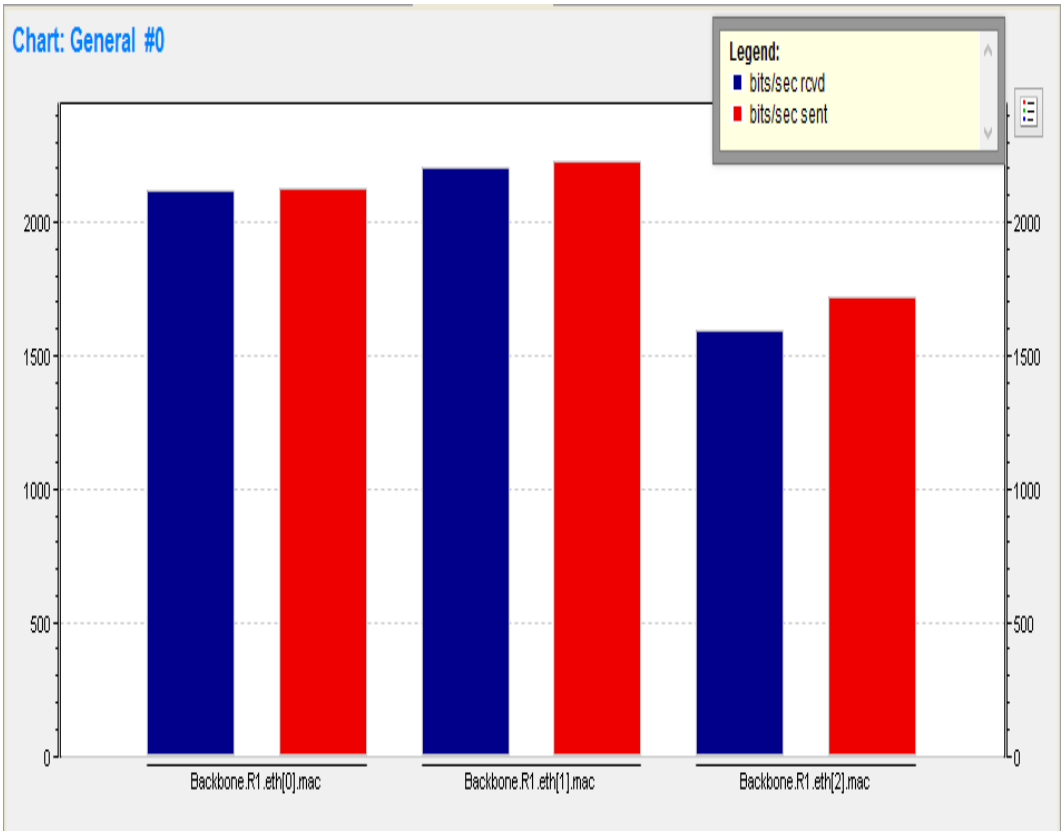
Gambar IV-4 Model Pengujian Topologi non Optimasi

Hasil serta nilai yang dihasilkan dari proses simulasi yang disajikan pada tabel-tabel dan gambar-gambar selanjutnya menunjukkan nilai dari ukuran suatu transfer data rata-rata pada jalur data utama yang digunakan saat proses komunikasi data berlangsung, hasil tersebut digunakan sebagai parameter yang dihasilkan untuk melakukan evaluasi terhadap *throughput* yang berkisar pada ketersediaan *bandwidth* yang cukup pada jalur komunikasi data utama ketika proses komunikasi data berlangsung yang dikirim dari pengirim menuju penerima paket data dalam jaringan komputer.

Transfer data rata-rata pada jalur utama yang terdapat pada *router* 1 serta nilai yang ditampilkan pada Gambar IV-5 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-5.

Tabel IV-5 Ukuran *Bandwidth* pada R1

Router	*Interface*	Ukuran *Bandwidth*
RI	Eth 0	Pengirim = 2125.9 *bytes* Penerima = 2116.3 *bytes*
	Eth 1	Pengirim = 2227.3 *bytes* Penerima = 2205.2 *bytes*
	Eth 2	Pengirim = 1721.6 *bytes* Penerima = 1596.4 *bytes*

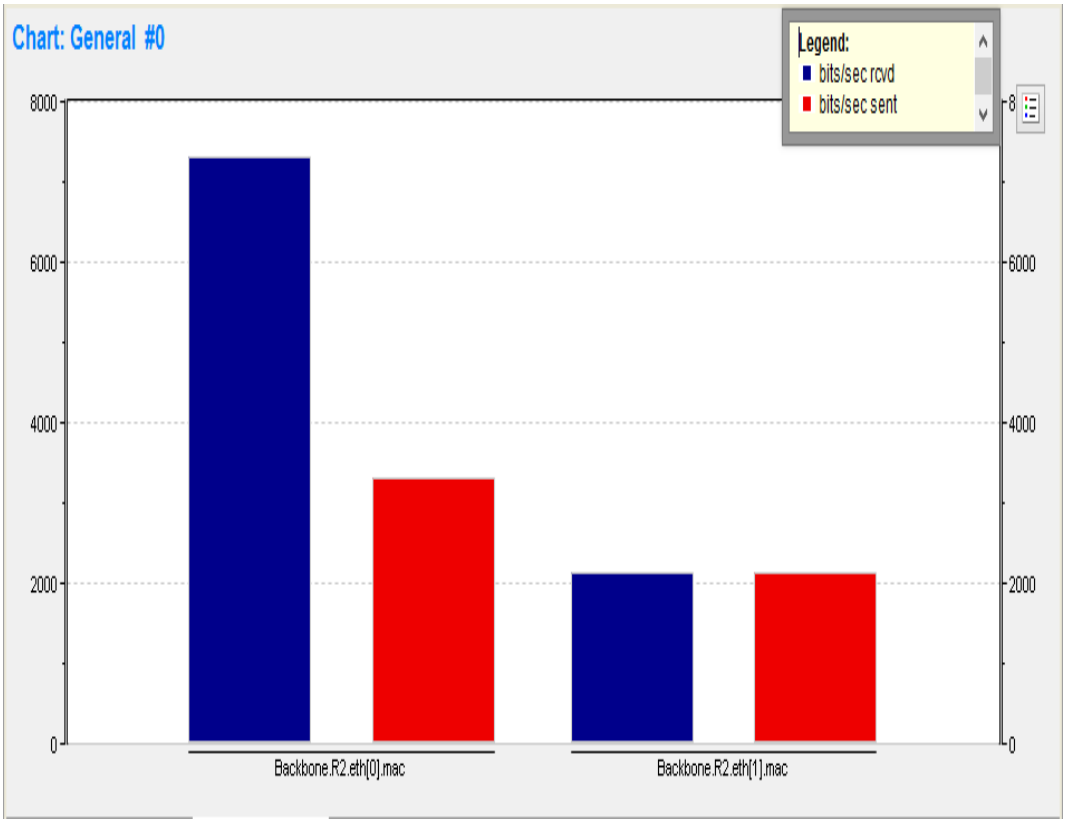


Gambar IV-5 *Throughput* Data pada Router 1

Transfer data pada jalur utama yang terdapat pada *router* 2 serta nilai yang ditampilkan pada Gambar IV-6 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-6.

Tabel IV-6 Ukuran *Bandwidth* pada R2

Router	*Interface*	Ukuran *Bandwidth*
R2	Eth 0	Pengirim = 3297.4 *bytes* Penerima = 7303.5 *bytes*
	Eth 1	Pengirim = 2116.4 *bytes* Penerima = 2125.9 *bytes*



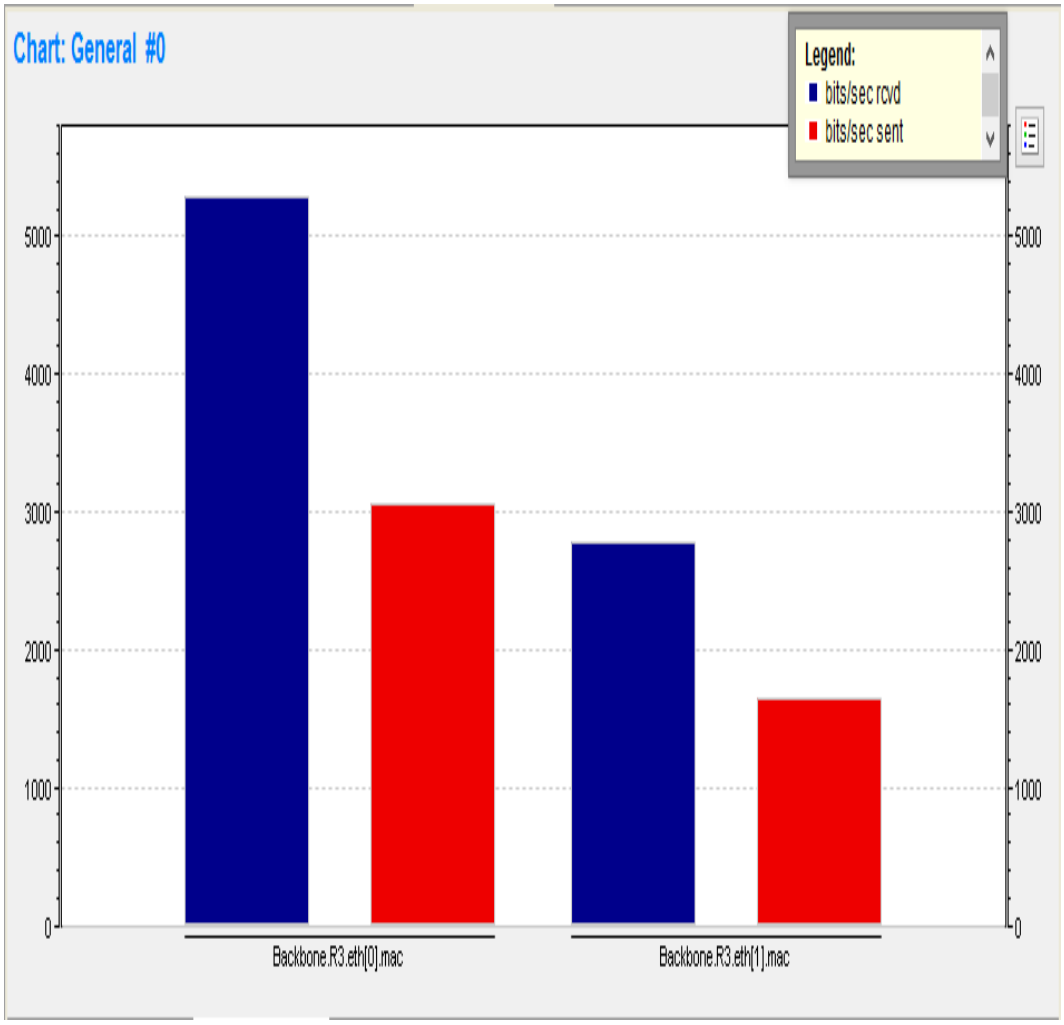
Gambar IV-6 *Throughput* Data pada Router 2

Transfer data pada jalur utama yang terdapat pada *router* 3 serta nilai yang ditampilkan pada Gambar IV-7 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-7.

Tabel IV-7 Ukuran *Bandwidth* pada R3

Router	*Interface*	Ukuran *Bandwidth*
R3	Eth 0	Pengirim = 3062.8 *bytes* Penerima = 5285.4 *bytes*

Router	*Interface*	Ukuran *Bandwidth*
	Eth 1	Pengirim = 1647.1 *bytes* Penerima = 2785.3 *bytes*



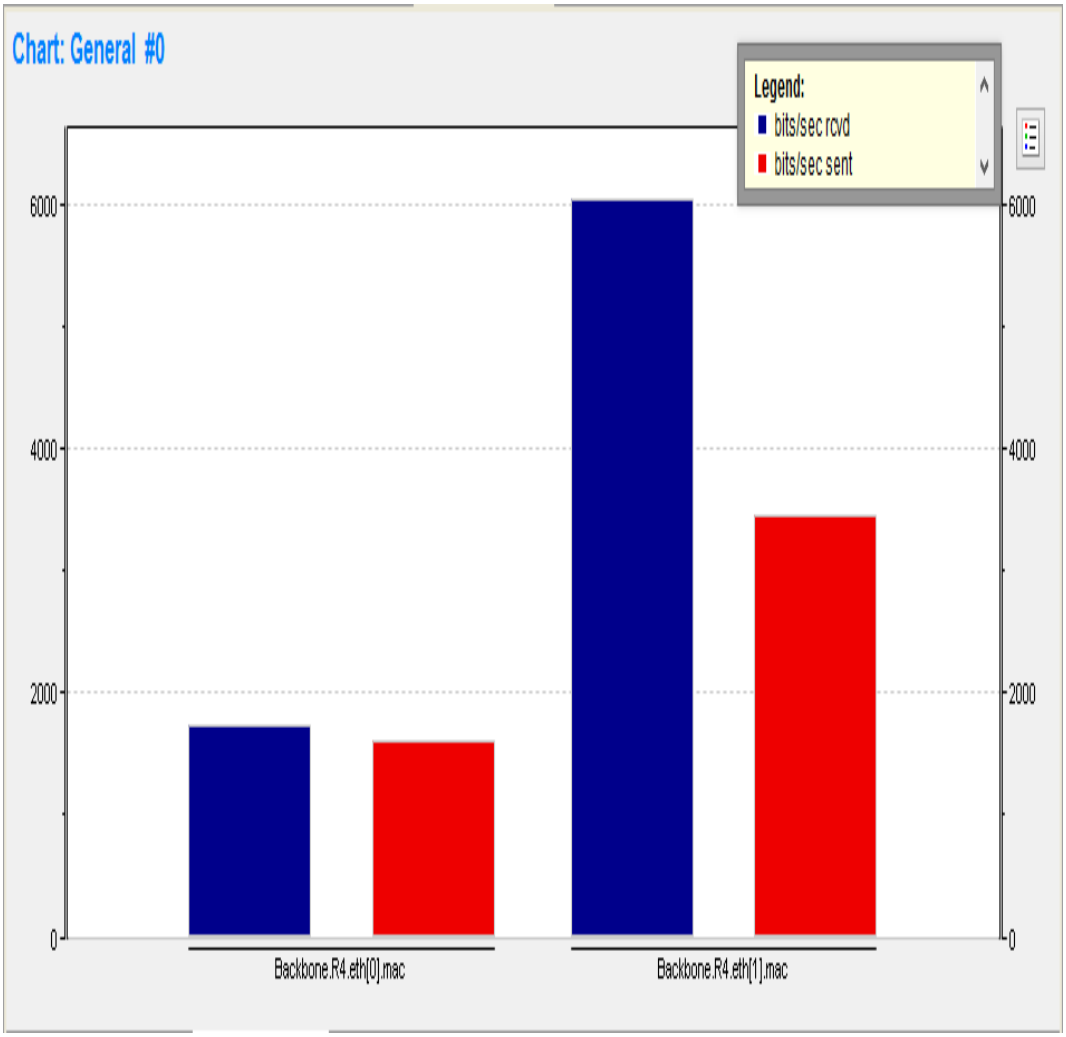
Gambar IV-7 *Throughput* Data pada Router 3

Transfer data pada jalur utama yang terdapat pada *router* 4 serta nilai yang ditampilkan pada Gambar IV-8 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-8.

Tabel IV-8 Ukuran *Bandwidth* pada R4

Router	*Interface*	Ukuran *Bandwidth*
R4	Eth 0	Pengirim = 1596.4 *bytes*

Router	*Interface*	Ukuran *Bandwidth*
		Penerima = 1721.6 *bytes*
	Eth 1	Pengirim = 3457.5 *bytes* Penerima = 6043.2 *bytes*

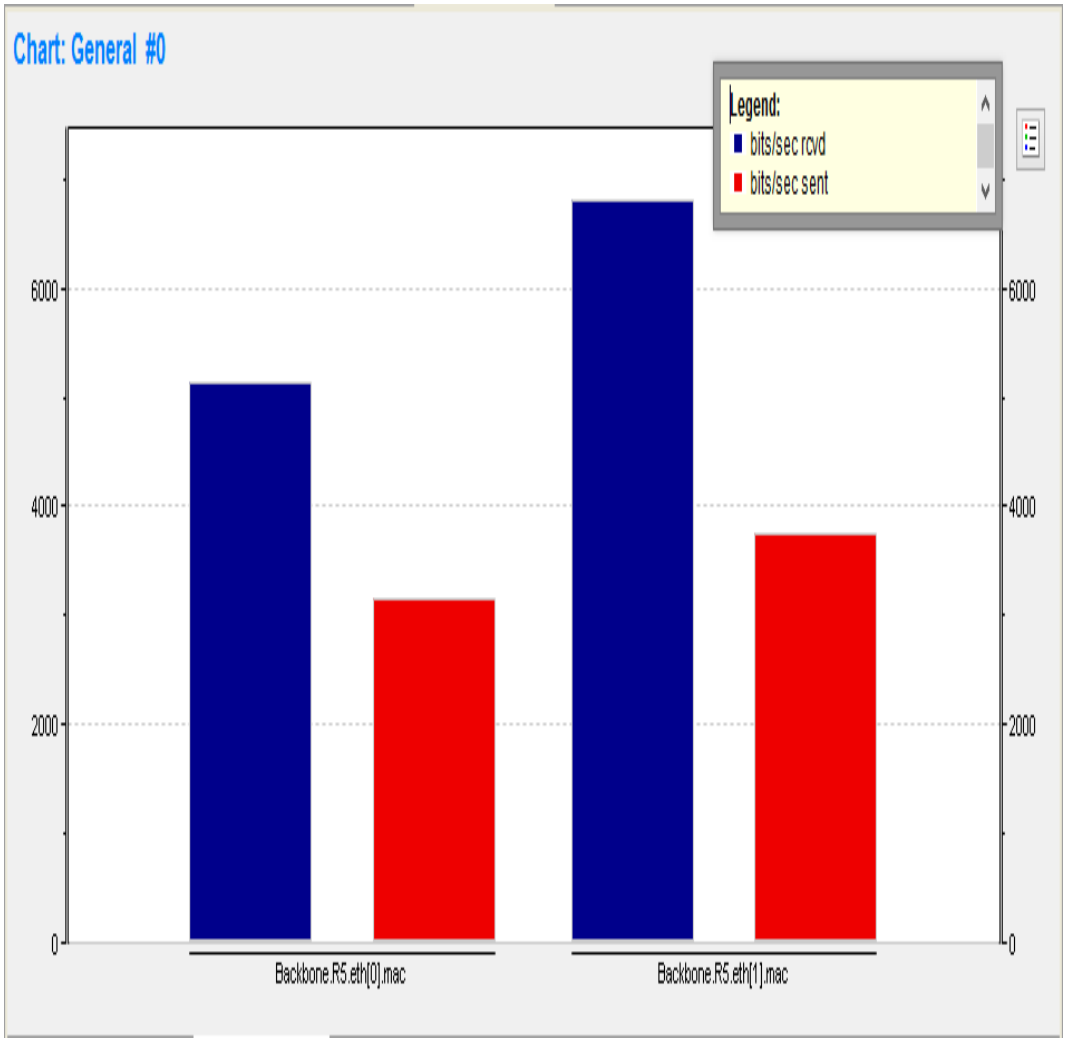


Gambar IV-8 *Throughput* Data pada Router 4

Transfer data pada jalur utama yang terdapat pada *router* 5 serta nilai yang ditampilkan pada Gambar IV-9 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-9.

Tabel IV-9 Ukuran *Bandwidth* pada R5

Router	*Interface*	Ukuran *Bandwidth*
R5	Eth 0	Pengirim = 3138.9 *bytes* Penerima = 5125.3 *bytes*
	Eth 1	Pengirim = 3742.9 *bytes* Penerima = 6807.3 *bytes*



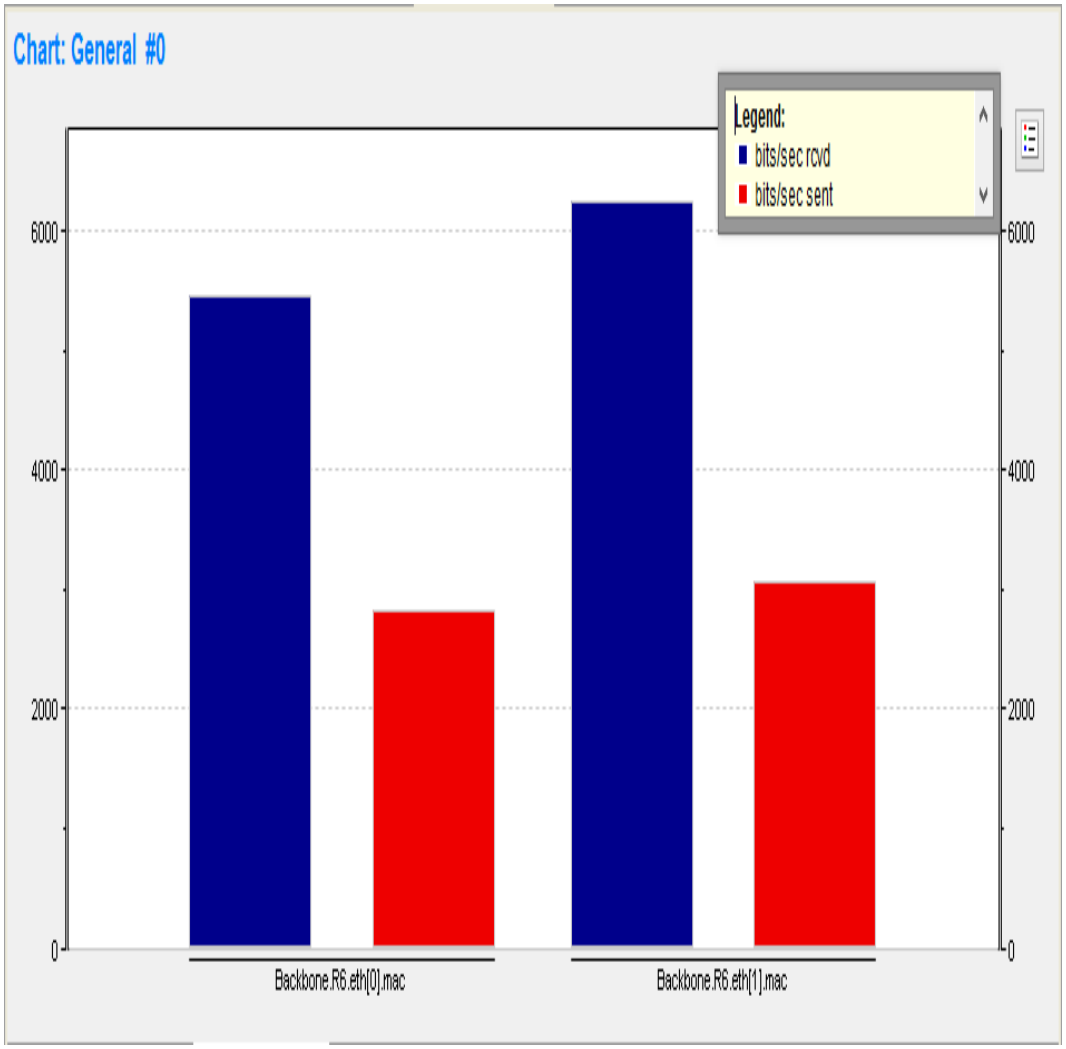
Gambar IV-9 *Throughput* Data pada Router 5

Transfer data pada jalur utama yang terdapat pada *router* 6 serta nilai yang ditampilkan pada Gambar IV-10 merupakan nilai dalam data skalar yang menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu

datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-10.

Tabel IV-10 Ukuran *Bandwidth* pada R6

Router	*Interface*	Ukuran *Bandwidth*
R6	Eth 0	Pengirim = 2823.4 *bytes* Penerima = 5459.8 *bytes*
	Eth 1	Pengirim = 3073.9 *bytes* Penerima = 6247.7 *bytes*



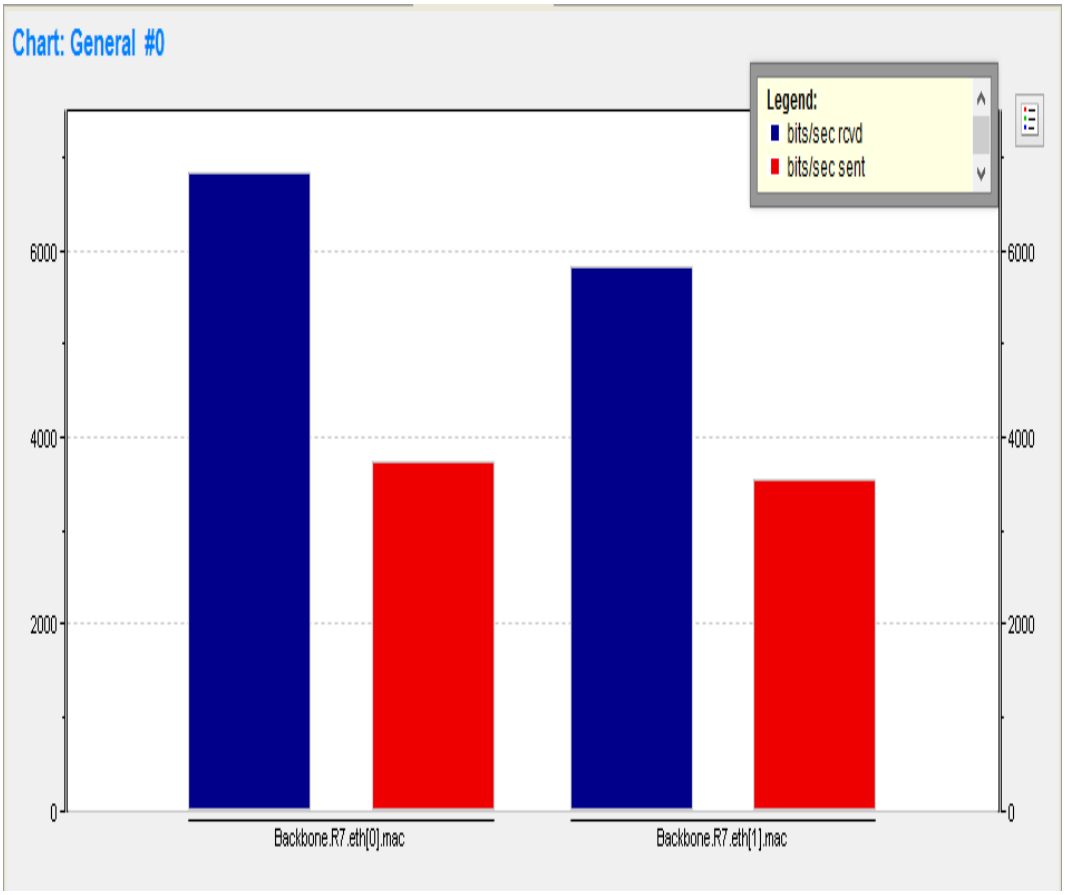
Gambar IV-10 *Throughput* Data pada Router 6

Transfer data pada jalur utama yang terdapat pada *router* 7 serta nilai yang ditampilkan pada Gambar IV-11 merupakan nilai dalam data skalar yang

menunjukkan *throughput* yang berkisar pada ketersediaan *bandwidth* dari suatu datagram yang dapat dilewatkan oleh suatu paket data saat terjadi transmisi data, untuk lebih jelasnya dapat dilihat pada Tabel IV-11.

Tabel IV-11 Ukuran *Bandwidth* pada R7

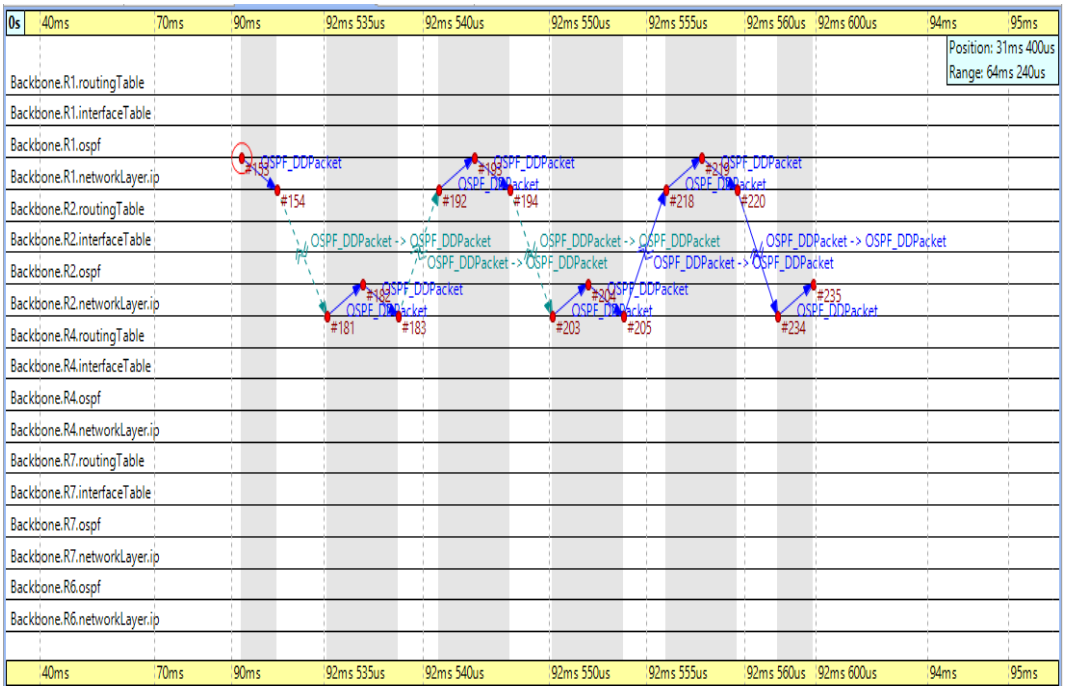
Router	*Interface*	Ukuran *Bandwidth*
R7	Eth 0	Pengirim = 3720.7 *bytes* Penerima = 6829.5 *bytes*
	Eth 1	Pengirim = 3536.8 *bytes* Penerima = 5829.2 *bytes*



Gambar IV-11 *Throughput* Data pada Router 7

Pengujian selanjutnya akan dilakukan pemodelan dan simulasi untuk mendapatkan suatu nilai yang dapat menunjukkan penggunaan *backbone* dan pemilihan *link* jalur utama yang digunakan untuk mengirimkan paket data saat proses komunikasi data dari pengirim menuju penerima pada suatu model jaringan

komputer yang menggunakan protokol routing OSPF, jalur utama yang digunakan dapat dilihat pada Gambar IV-12.

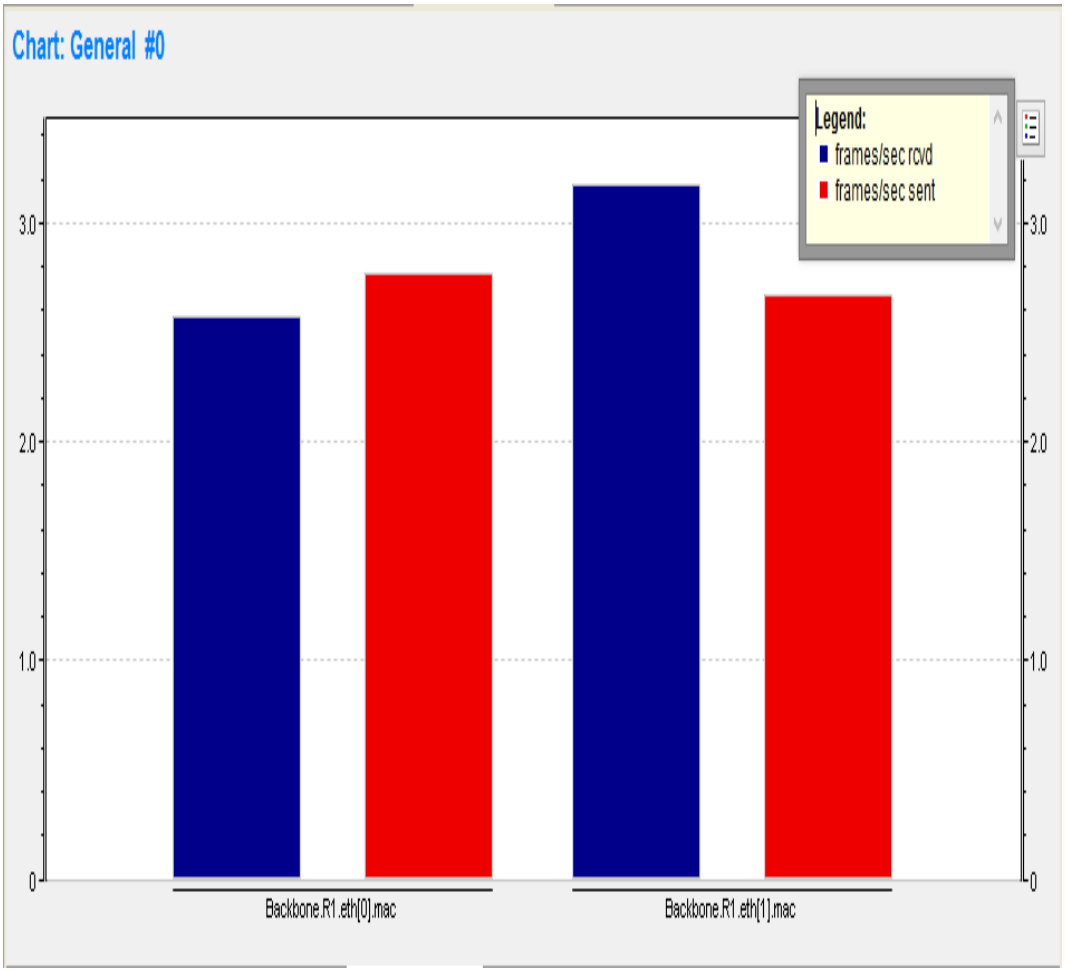


Gambar IV-12 Penggunaan Jalur Utama saat transmisi data

Hasil yang ditunjukkan pada Gambar IV-12 menunjukkan penggunaan *backbone* dan pemilihan jalur *link* utama untuk proses komunikasi data yang dilakukan oleh protokol *routing* OSPF pada model topologi jaringan non optimasi dibebankan hanya pada salah satu *link* jalur utama saja, mekanisme pendistribusian jalur yang dilakukan pada model topologi jaringan non optimasi oleh protokol *routing* OSPF yang hanya melakukan pendistribusian jalur pada satu jalur utama saja akan berakibat pada terjadinya kongesti ketika terjadi proses komunikasi data berlangsung, hal ini akan berakibat terjadinya kemungkinan perubahan kondisi beban data serta penurunan kualitas dari ukuran *bandwidth*.

Pengujian selanjutnya akan dilakukan simulasi transmisi data berupa pengiriman pesan suara (*voice*) sesuai skenario pengujian kedua untuk melihat terjadinya kongesti serta kondisi beban data yang harus dilayani pada jalur utama saat proses komunikasi berlangsung pada model topologi non optimasi, hasil yang

disajikan berupa data skalar pada *router* 1 pada jalur utama seperti yang terlihat pada Gambar IV-13.



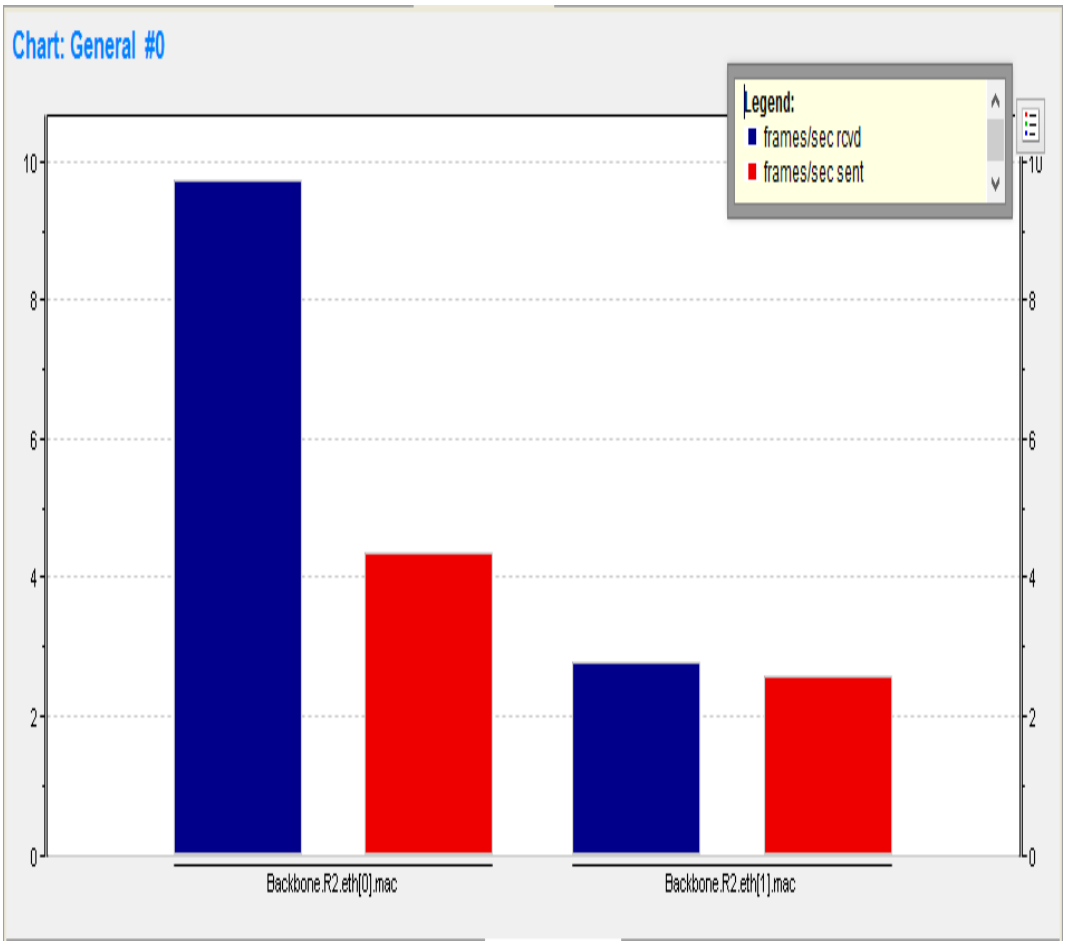
Gambar IV-13 Beban Paket Data Suara pada R1

Nilai untuk setiap ukuran *frames* yang dilewatkan pada jalur yang digunakan saat pengiriman paket data suara untuk setiap *interfaces* pada *router* 1 dapat dilihat pada Tabel IV-12.

Tabel IV-12 Beban Paket Data Suara pada R1

Router	*Interface*	Ukuran *Frames*
R1	Eth 0	Pengirim = 2774 *kbytes* Penerima = 2576 *kbytes*
	Eth 1	Pengirim = 2675 *kbytes* Penerima = 3170 *kbytes*

Kondisi beban data pada jalur utama saat proses komunikasi berlangsung pada model topologi non optimasi, hasil yang disajikan berupa data skalar pada *router* 2 pada jalur utama seperti yang terlihat pada Gambar IV-14.



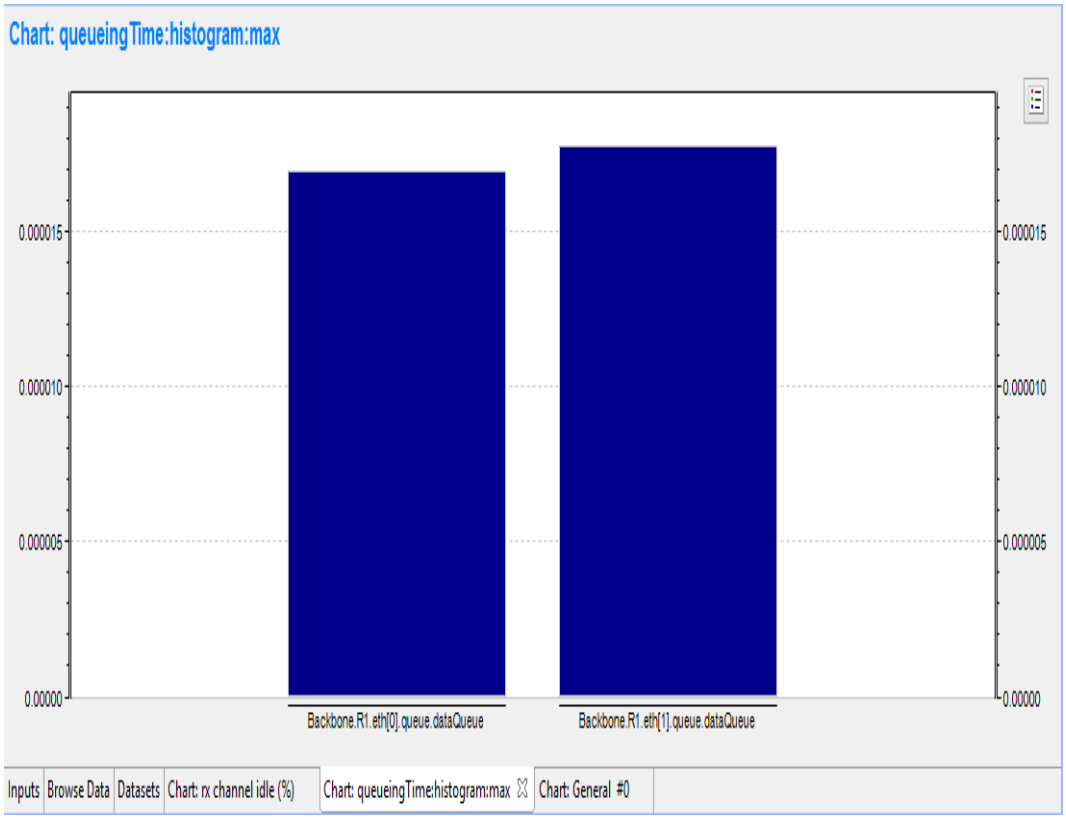
Gambar IV-14 Beban Paket Data Suara pada R2

Nilai untuk setiap ukuran *frames* yang dilewatkan pada jalur yang digunakan saat pengiriman paket data suara untuk setiap *interfaces* pada *router* 2 dapat dilihat pada Tabel IV-13.

Tabel IV-13 Beban Paket Data Suara pada R2

Router	*Interface*	Ukuran *Frames*
R2	Eth 0	Pengirim = 4359 *kbytes* Penerima = 9710 *kbytes*
	Eth 1	Pengirim = 2576 *kbytes* Penerima = 2774 *kbytes*

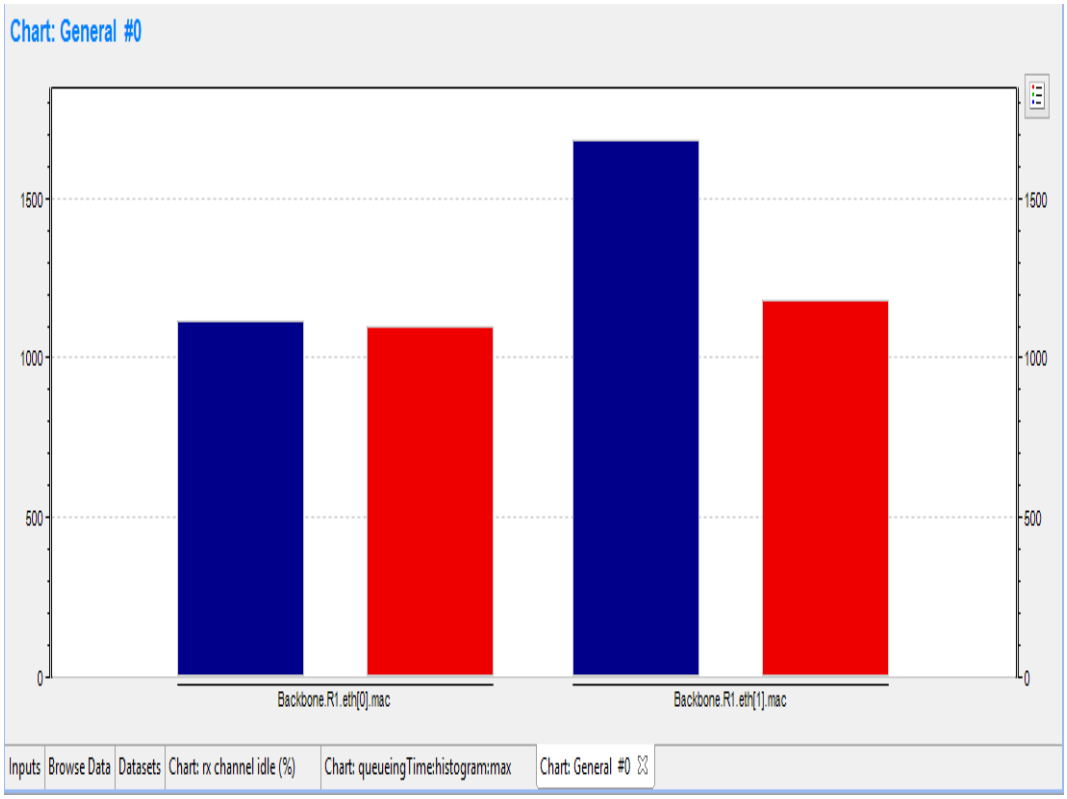
Hasil pengujian selanjutnya pada model topologi jaringan non optimasi yang menggunakan protokol *routing* OSPF akan dilakukan penilaian terhadap tingkat *delay* yang terjadi pada saat proses komunikasi data berlangsung, *delay* merupakan waktu yang dibutuhkan untuk mengirimkan suatu paket data dari pengirim menuju penerima pada jaringan komputer. Pengujian akan dilakukan dengan melakukan pengiriman paket ICMP pada interval waktu pengiriman 0.1 detik (*second*). Hasil yang di dapat dari simulasi dapat dilihat Gambar IV-15. Nilai yang di dapat saat proses simulasi dengan melakukan test koneksi ICMP *ports* (ping) pada interval waktu pengiriman 0.1 detik *delay* maksimum yang di dapat 0.00018ms atau sebesar $2x10^{-8}$ detik. Nilai *delay* pada model protokol *routing* OSPF yang di dapat sangat kecil, hal ini dikarenakan proses komunikasi yang terjadi hanya terjadi proses transmisi data.



Gambar IV-15 Delay saat pengiriman paket ICMP

Pengujian selanjutnya akan dilakukan simulasi berupa pengiriman paket ICMP bersamaan dengan melakukan simulasi pengiriman suatu paket data suara

sebesar 64 *bytes* untuk mengetahui nilai *delay* yang terjadi pada saat proses pengiriman data dalam waktu yang bersamaan. Standar untuk *delay* maksimum yang direkomendasikan oleh *International Telecommunication Union* (ITU) untuk mengirimkan suatu paket data suara dengan kualitas data terbaik adalah 0.15s, dan yang masih diterima oleh pengguna adalah sebesar 250ms. Hasil *delay* yang terjadi pada saat dilakukan simulasi dengan melakukan proses pengiriman data dalam waktu yang bersamaan dapat dilihat pada Gambar IV-16.



Gambar IV-16 Delay pada saat terjadi proses komunikasi bersamaan

Hasil yang disajikan pada Gambar IV-16 menunjukkan tingkat *delay* yang terjadi pada saat terjadi proses komunikasi data yang bersamaan mengalami peningkatan, nilai untuk tingkat *delay* yang dihasilkan mencapai 1700ms atau 1.7 detik melebihi tingkat *delay* yang sudah ditetapkan oleh ITU yaitu sebesar 150ms. Tingkat *delay* yang mencapai melebihi batas nilai standar *delay* ini diakibatkan adanya antrian data pada komunikasi di model jaringan dengan menggunakan protokol *routing* OSPF. Antrian data yang terjadi karena protokol *routing* OSPF

menggunakan satu jalur komunikasi data saja, sehingga paket data yang akan dikirimkan akan menunggu sampai paket data yang dikirimkan diawal sampai ke tujuan terlebih dahulu, baru setelah itu paket data yang mengantri setelahnya akan dikirimkan ke tujuan.

Pengujian yang kedua akan dilakukan simulasi pada model topologi optimasi yang menerapkan model MPLS *traffic engineering*, *source code* yang diterapkan di setiap node LSR (*Label Switching Router*) sebagai berikut:

```
<?xml version="1.0"?>
<sessions>
  <session>
    <endpoint>host3</endpoint>
    <tunnel_id>1</tunnel_id>

    <setup_pri>1</setup_pri>
    <holding_pri>1</holding_pri>

    <paths>
      <path>
        <lspid>100</lspid>

        <bandwidth>400000</bandwidth>
        <route>
          <node>LSR1%routerId</node>
          <node>LSR3%routerId</node>
          <node>LSR7%routerId</node>
          <node>LSR5%routerId</node>
        </route>

        <permanent>true</permanent>
        <color>100</color>
      </path>
    </paths>
  </session>
  <session>
    <endpoint>host4</endpoint>
    <tunnel_id>2</tunnel_id>

    <paths>
      <path>
        <lspid>200</lspid>

        <bandwidth>400000</bandwidth>
        <route>
          <node>LSR1%routerId</node>
          <node>LSR2%routerId</node>
          <node>LSR4%routerId</node>
          <node>LSR5%routerId</node>
        </route>

        <permanent>true</permanent>
```

```
          <color>200</color>
        </path>
      </paths>
    </session>
</sessions>
```

Source code skenario pengujian kedua yang akan diterapkan untuk memodelkan dan mensimulasikan topologi optimasi dengan model MPLS (*Multi Protocol Label Switching*) sebagai berikut:

```
[General]
network = RSVPTE4
sim-time-limit = 3600s

total-stack = 64MiB
tkenv-plugin-path = ../../../etc/plugins

**.host1.numUdpApps = 1
**.host1.udpApp[*].typename = "UDPBasicApp"
**.host1.udpApp[0].localPort = 100
**.host1.udpApp[0].destPort = 100
**.host1.udpApp[0].messageLength = 128 bytes

**.host1.udpApp[0].sendInterval = 0.10s
**.host1.udpApp[0].destAddresses = "host3"

**.host2.numUdpApps = 1
**.host2.udpApp[*].typename = "UDPBasicApp"
**.host2.udpApp[0].localPort = 100
**.host2.udpApp[0].destPort = 100
**.host2.udpApp[0].messageLength = 128 bytes

**.host2.udpApp[0].sendInterval = 0.05s
**.host2.udpApp[0].destAddresses = "host4"

**.host{3..4}.numUdpApps = 1
**.host{3..4}.udpApp[*].typename = "UDPSink"
**.host{3..4}.udpApp[0].localPort = 100

**.lossRate.result-recording-modes = all
**.channel.throughput.result-recording-modes = all

# ip config
**.host1.routingFile = "host1.rt"
**.host2.routingFile = "host2.rt"
**.host3.routingFile = "host3.rt"
**.host4.routingFile = "host4.rt"
**.host5.routingFile = "host5.rt"

# LSR configuration

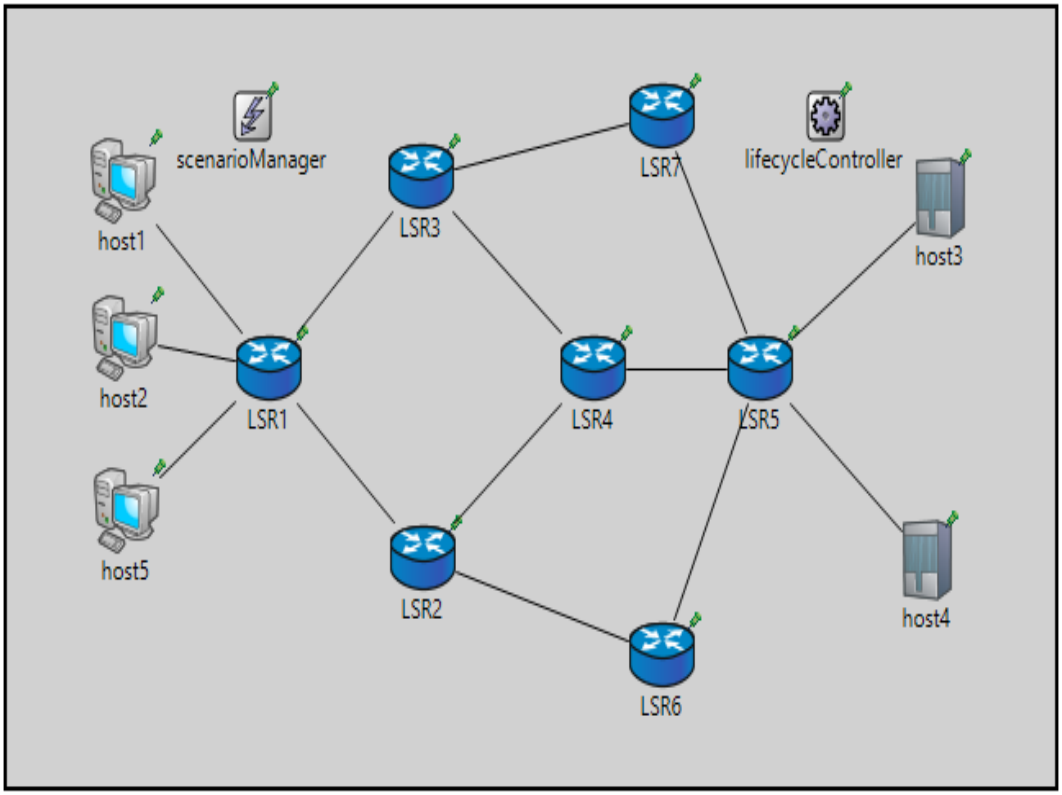
**.LSR1.classifier.config = xmldoc("LSR1_fec.xml")
**.LSR1.rsvp.traffic = xmldoc("LSR1_rsvp.xml")
```

```
**.LSR*.rsvp.helloInterval = 360s
**.LSR*.rsvp.helloTimeout = 100s

**.LSR1.routingFile = "LSR1.rt"
**.LSR2.routingFile = "LSR2.rt"
**.LSR3.routingFile = "LSR3.rt"
**.LSR4.routingFile = "LSR4.rt"
**.LSR5.routingFile = "LSR5.rt"
**.LSR6.routingFile = "LSR6.rt"
**.LSR7.routingFile = "LSR7.rt"
# NIC configuration
**.ppp[*].queueType = "DropTailQueue" # in routers
**.ppp[*].queue.frameCapacity = 100  # in routers

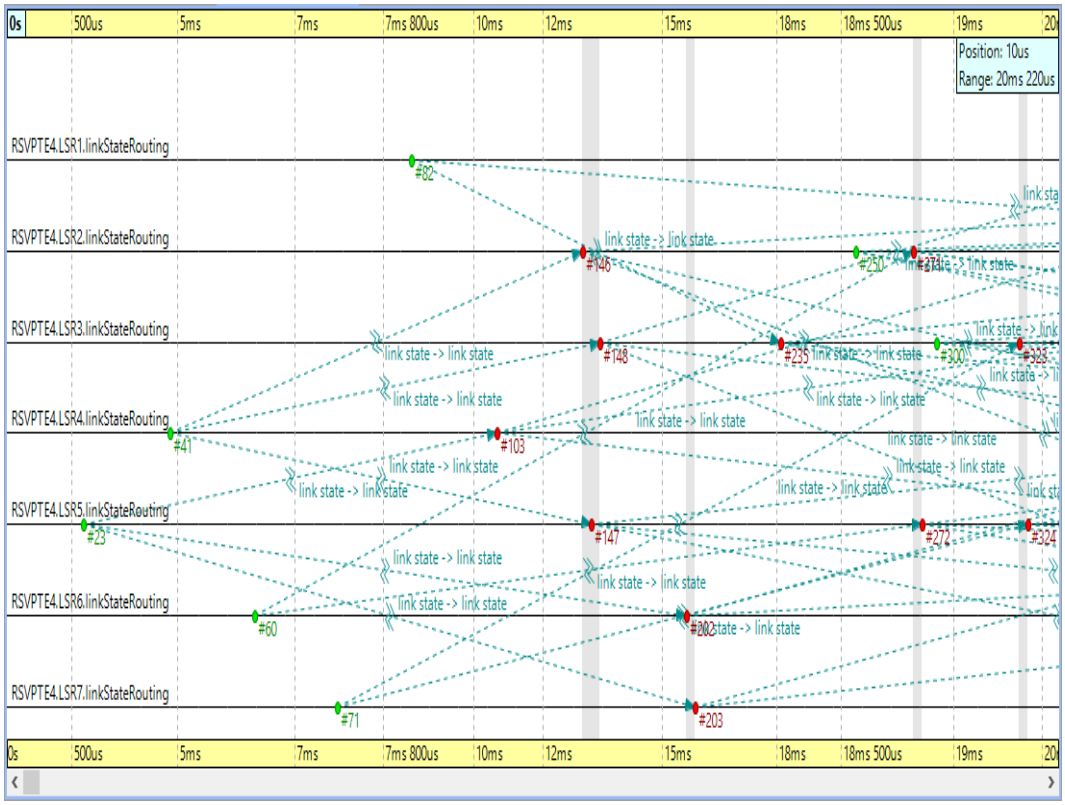
# scenario
**.scenarioManager.script = xmldoc("scenario.xml")
```

Rancangan topologi optimasi pada model MPLS yang digunakan dengan skenario utama yang sudah dijelaskan sebelumnya, hanya saja yang membedakan adalah penggunaan LSR (*Label Switching Router*) sebagai node pada model yang digunakan seperti yang terlihat pada Gambar IV-17.



Gambar IV-17 Model Pengujian Topologi Optimasi

Pengujian selanjutnya akan dilakukan simulasi dan pemodelan yang digunakan untuk mendapatkan informasi yang menunjukkan penggunaan jalur *link* utama komunikasi data dan distribusi suatu informasi yang dilakukan oleh LSR (*Label Switching Router*) pada model topologi jaringan optimasi dengan MPLS *traffic engineering* dalam menentukan serta mendistribusikan paket data dari pengirim menuju penerima pada jalur utama yang digunakan ketika ada proses komunikasi data. Pengujian yang dilakukan untuk menguji dan hasil yang didapatkan kemudian dibandingkan dengan hasil yang didapatkan dari pengujian simulasi pada topologi non optimasi yang menggunakan protokol *routing* OSPF. Hasil yang disajikan pada Gambar IV-18 merupakan informasi penggunaan *link* jalur utama yang digunakan pada saat proses komunikasi data berlangsung.

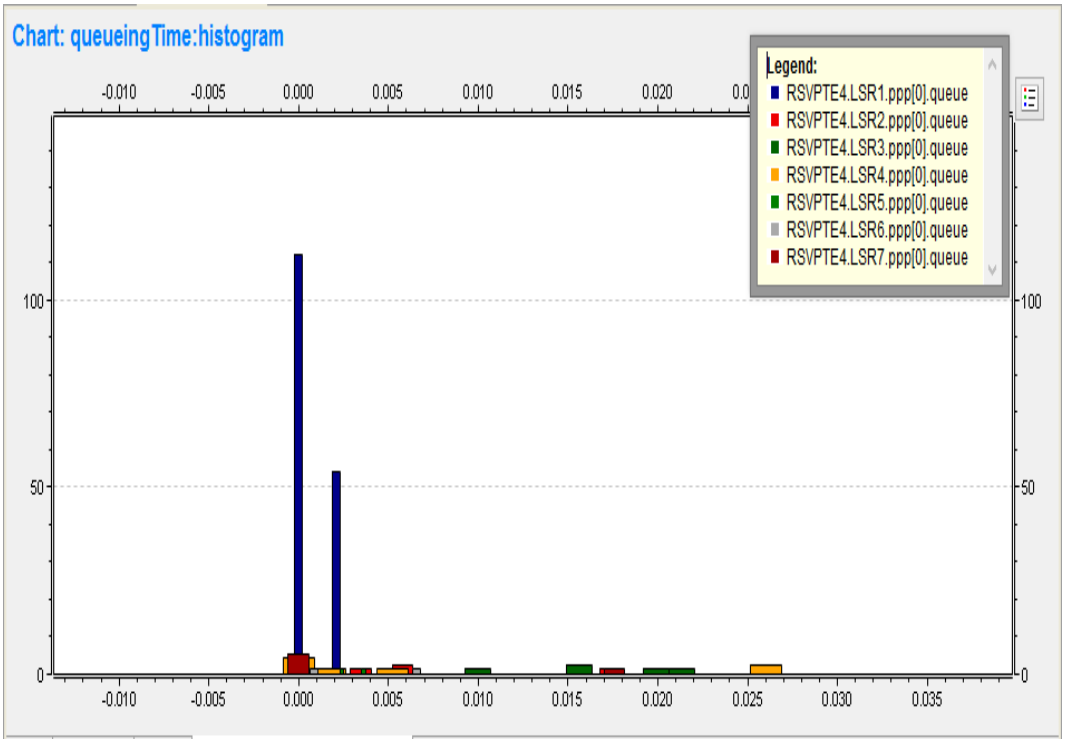


Gambar IV-18 Penggunaan Jalur pada Model MPLS

Hasil yang didapatkan pada Gambar IV-18 menunujukkan adanya mekanisme distribusi informasi jalur yang merata sebelum dilakukannya suatu pengiriman paket data menuju penerima berupa *broadcast information* yang bertujuan untuk mengetahui ukuran *bandwidth* serta kondisi beban pada jalur

yang akan digunakan untuk transmisi data. *Traffic engineering* digunakan untuk mengatur aliran trafik data dalam jaringan untuk mengoptimalkan penggunaan *bandwidth* yang berlebihan pada jaringan, untuk memilih rute dan memindahkan trafik dari jalur dengan tingkat kongesti lebih besar ke jalur dengan tingkat kongesti lebih kecil.

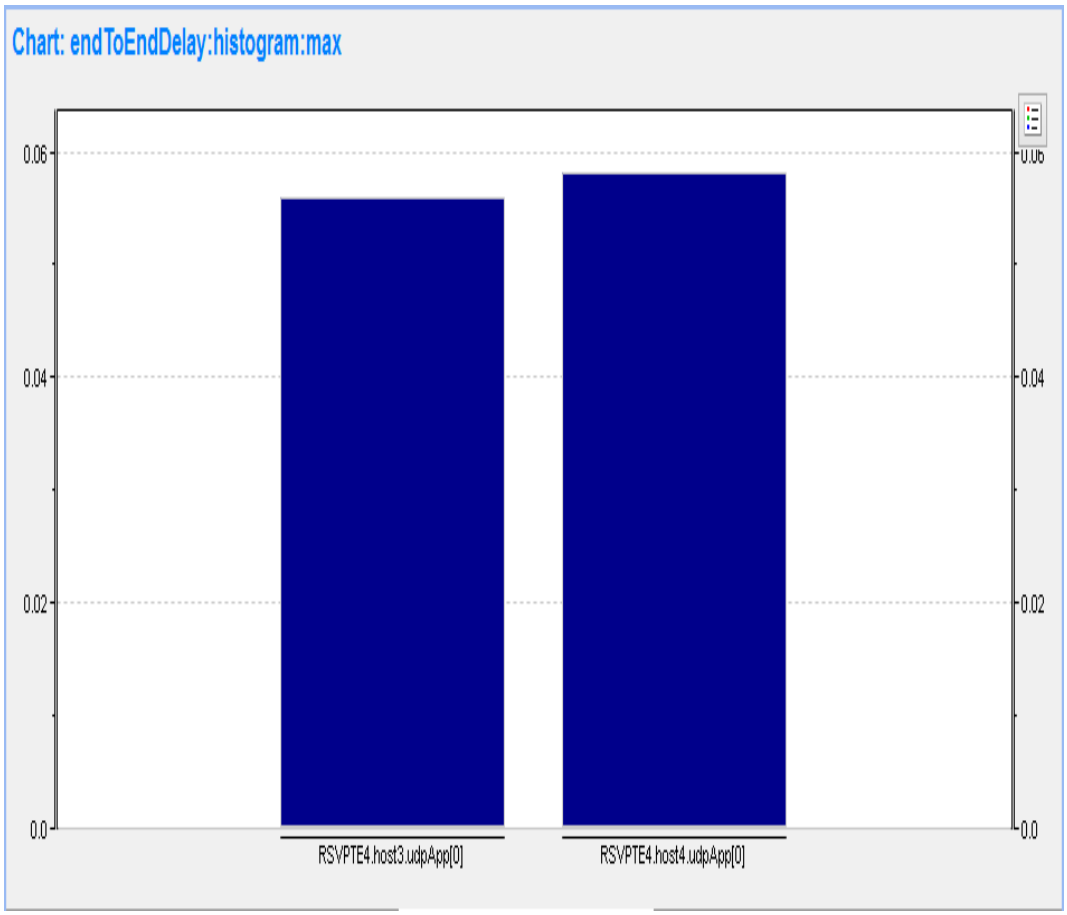
Hasil yang didapatkan pada simulasi selanjutnya adalah data histogram yang menunjukkan waktu tunggu (*delay*) yang terjadi ketika dilakukan pengujian dengan melakukan pengiriman berupa data suara (*voice*), ketika proses pertama kali data suara akan dikirimkan jelas terlihat waktu tunggu ketika data akan dikirimkan sangat tinggi, akan tetapi pada iterasi selanjutnya waktu tunggu yang terjadi mengalami penurunan yang diakibatkan adanya pembagian pesan data yang dilakukan oleh LSR sesuai dengan kondisi disetiap beban jalur. Hasil terlihat pada Gambar IV-19.



Gambar IV-19 Delay saat pengiriman paket data suara

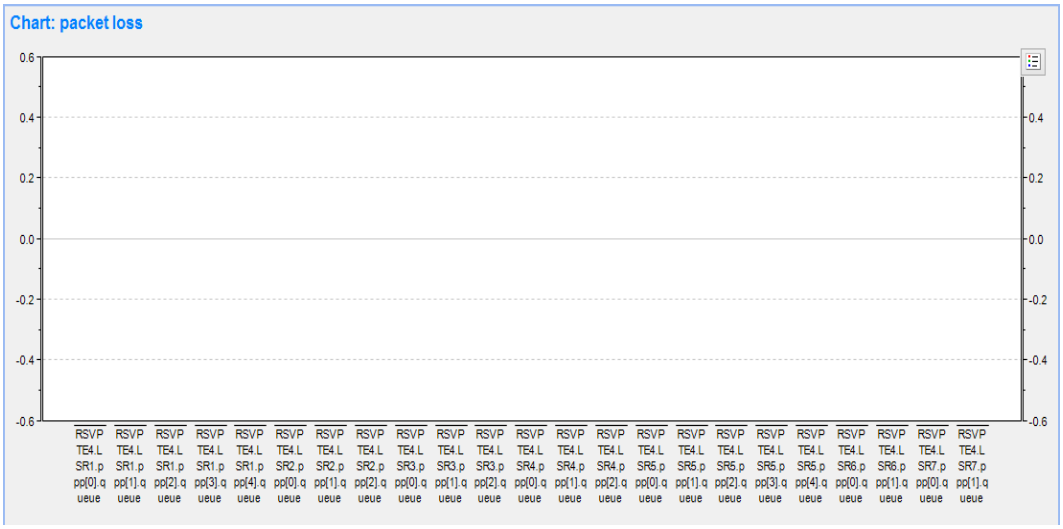
Pengujian selanjutnya akan dilakukan pemodelan dan simulasi berupa pengiriman paket ICMP bersamaan dengan melakukan simulasi pengiriman suatu paket data suara sebesar 64 *bytes* untuk mengetahui nilai *delay* yang terjadi pada

saat proses pengiriman data dalam waktu yang bersamaan pada model topologi optimasi dengan model multi protokol label *switching* (MPLS), data yang dihasilkan dapat dilihat pada Gambar IV-20.



Gambar IV-20 Delay saat pengiriman paket data bersamaan

Hasil pengujian berikutnya adalah data yang menunjukkan hasil simulasi dan pemodelan topologi optimasi dengan model MPLS yang menunjukkan tingkat *packet loss* yang terjadi pengiriman berupa paket data suara (*voice*), *packet loss* yang terjadi disebabkan karena adanya proses dekapsulasi data yang dilakukan oleh LSR menjadi beberapa ukuran yang disesuaikan dengan kapasitas jalur komunikasi data yang akan dilalui sebelum data dikirimkan. Nilai dari pengujian yang dihasilkan ditampilkan dalam bentuk vektor dapat dilihat pada Gambar IV-21.



Gambar IV-21 Packet Loss

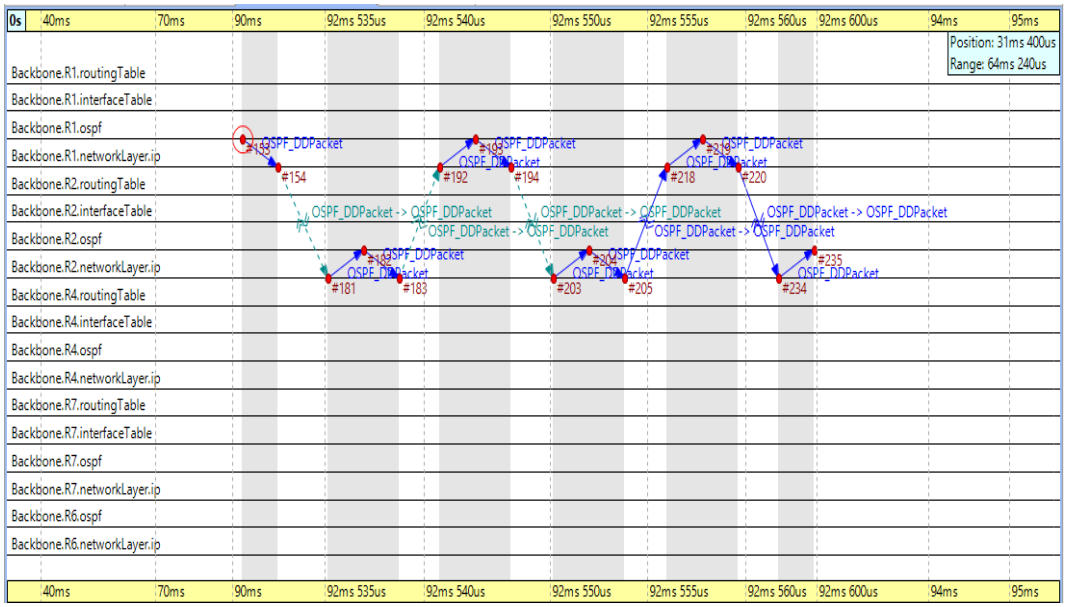
IV.3. Evaluasi dari Pengujian

Evaluasi yang didapatkan dari hasil pengujian yang telah disimulasikan sesuai dengan skenario utama diantaranya:

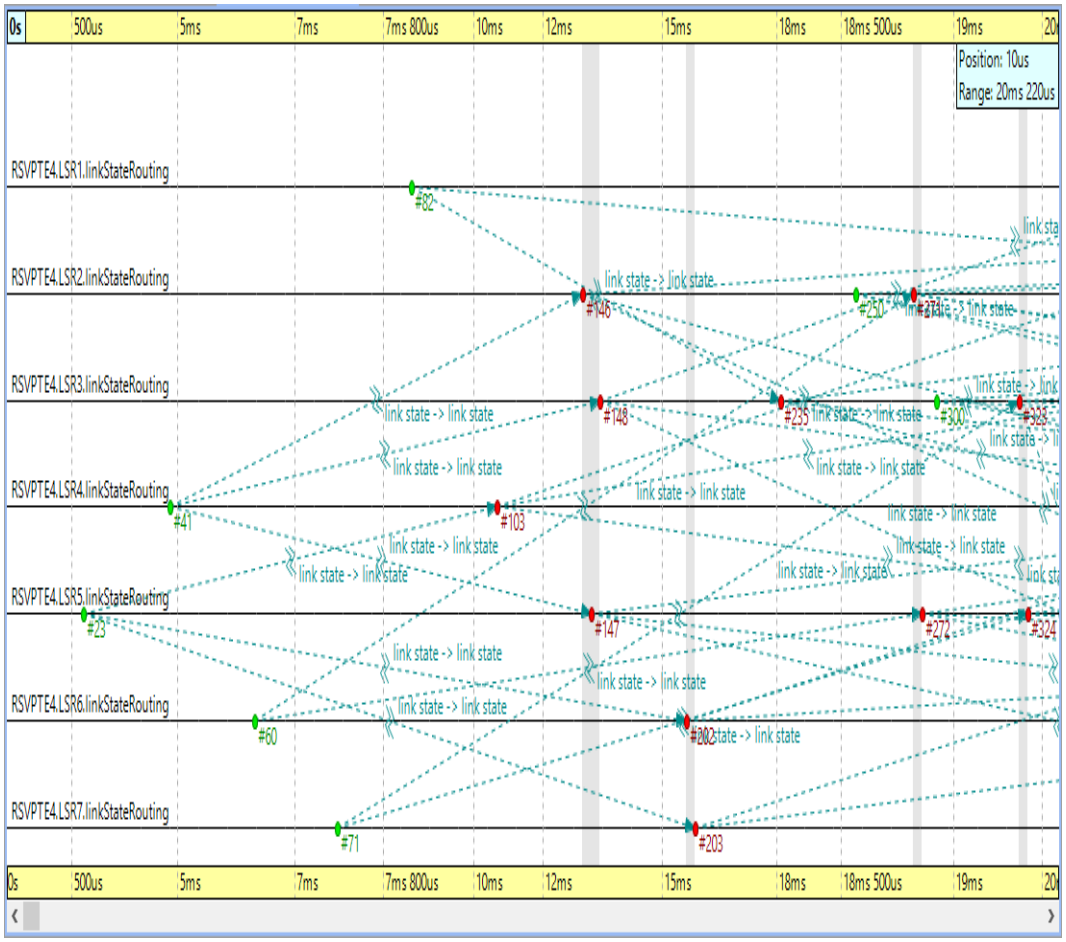
1. Penggunaan Jalur

   Penggunaan jalur dan pemilihan rute utama yang digunakan pada model non optimasi terlihat kurang efisien karena jalur utama yang digunakan hanya berpusat pada satu jalur komunikasi utama berdasarkan pada kualitas *bandwidth* yang besar tanpa memperhatikan adanya jalur utama lainnya dengan tingkat kepadatan data yang relatif kecil.

   Hasil dari pengujian simulasi pada model topologi jaringan optimasi menunjukkan tingkat konvergensi pada jalur relatif kecil, karena mekanisme dalam pemilihan jalur tidak berpusat pada satu jalur komunikasi utama dengan memperhatikan ukuran *bandwidth*, tetapi lebih memperhatikan pada kondisi beban jalur yang kosong, sehingga dapat tercipta keseimbangan pada jalur walaupun tingkat komunikasi sedang tinggi. Perbandingan penggunaan jalur yang digunakan untuk mengirimkan paket data saat terjadi proses komunikasi dapat dilihat pada Gambar IV-22 merupakan hasil pengujian pada topologi jaringan sebelum dilakukan optimasi, sedangkan untuk penggunaan jalur pada topologi jaringan setelah dilakukan optimasi dapat dilihat pada Gambar IV-23.



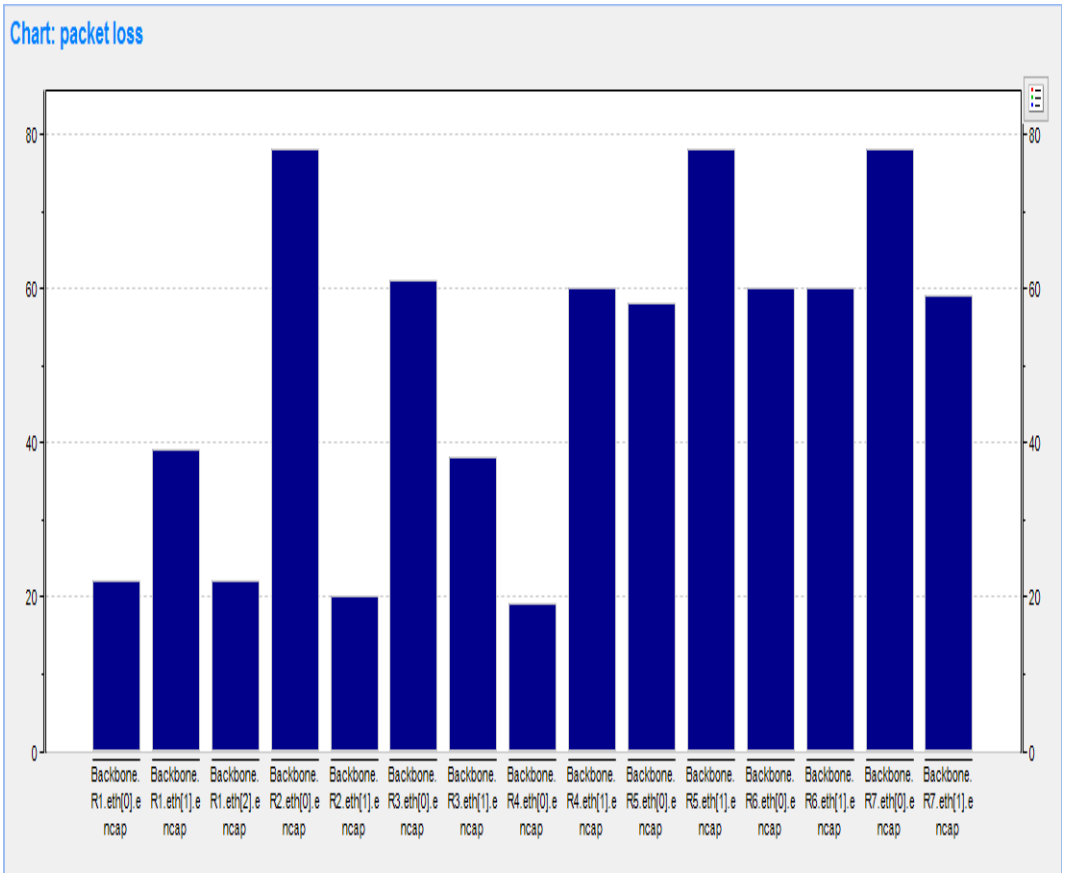
Gambar IV-22 Penggunaan Jalur sebelum Optimasi



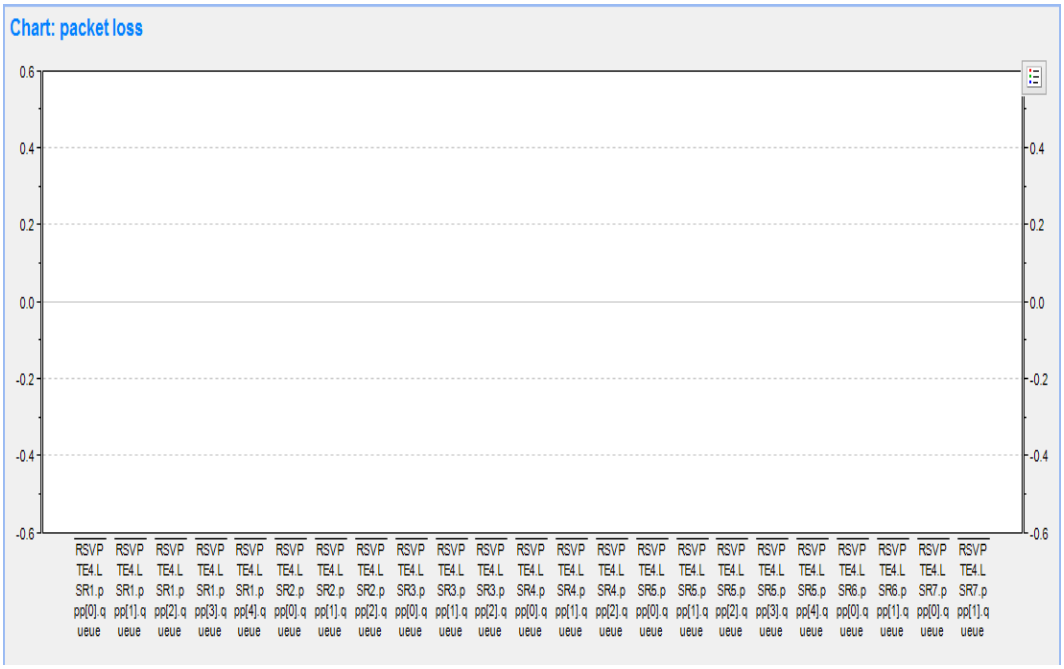
Gambar IV-23 Penggunaan Jalur setelah Optimasi

2. Kondisi Beban Jalur

Kondisi beban jalur yang disimulasikan pada model topologi jaringan non optimasi kurang stabil, karena rute trafik pemilihan jalur hanya satu jalur saja, yang ditentukan berdasarkan pada jalur terpendek dari satu titik ke titik lainnya dengan menggunakan *hop* terkecil untuk membentuk suatu informasi pada tabel routing tanpa adanya pembangian jalur utama (*load balancing*) untuk mengirimkan paket data. Jalur yang digunakan akan terus dipertahankan sampai jalur utama tersebut mengalami putus hubungan. Hal ini akan mengakibatkan terjadinya tabrakan data yang dapat menyebabkan nilai *packet loss* pada transmisi data tinggi. Kondisi jalur beban jalur pada topologi jaringan non optimasi dengan nilai *packet loss* dapat dilihat pada Gambar IV-24, sedangkan kondisi beban jalur pada topologi jaringan optimasi dengan nilai *packet loss* dapat dilihat pada Gambar IV-25.



Gambar IV-24 Packet loss pada topologi non Optimasi



Gambar IV-25 Packet loss pada topologi Optimasi

Evaluasi pengujian selanjutnya akan di lakukan perbandingan nilai dari seluruh pengujian pada topologi sebelum dilakukan optimasi dengan nilai yang telah ditentukan oleh standar *International Telecommunication Union* (ITU) [9] dapat dilihat pada Tabel IV-14.

Tabel IV-14 Standar Nilai Packet loss ITU

Kategori	Tigkat *Packet loss*
Sangat Bagus	0 %
Bagus	1 – 3 %
Sedang	4 – 15 %
Buruk	16 – 25 %

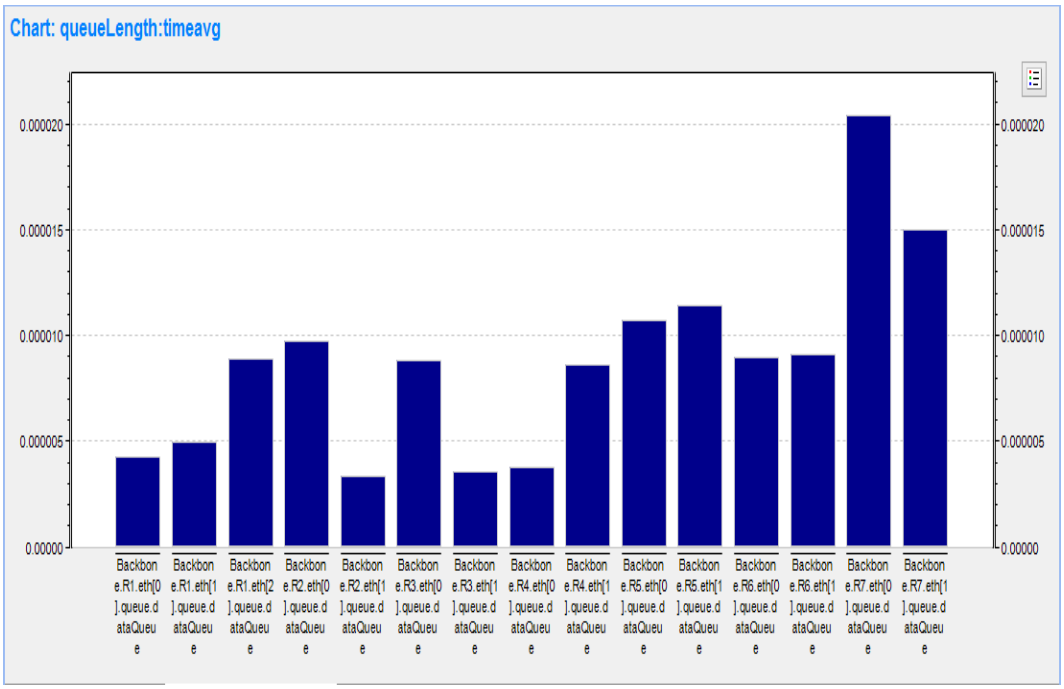
Hasil yang dapat disimpulkan bahwa *packet loss* yang terjadi pada topologi jaringan non optimasi sangat tinggi, dengan demikian tidak disarankan diterapkan pada jaringan multi jalur, karena rata dari nilai *packet loss* yang terjadi pada interval waktu selama 0.1 detik dengan paket data yang dikirimkan berupa paket data suara (*voice*) mencapai 78%. Sedangkan pada topologi setelah dilakukan optimasi tidak mengalami *packet loss* nilai 0%. Perbandingan nilai *packet loss* pada topologi non optimasi dengan topologi optimasi dapat dilihat pada Tabel IV-15.

Tabel IV-15 Perbandingan nilai Packet loss

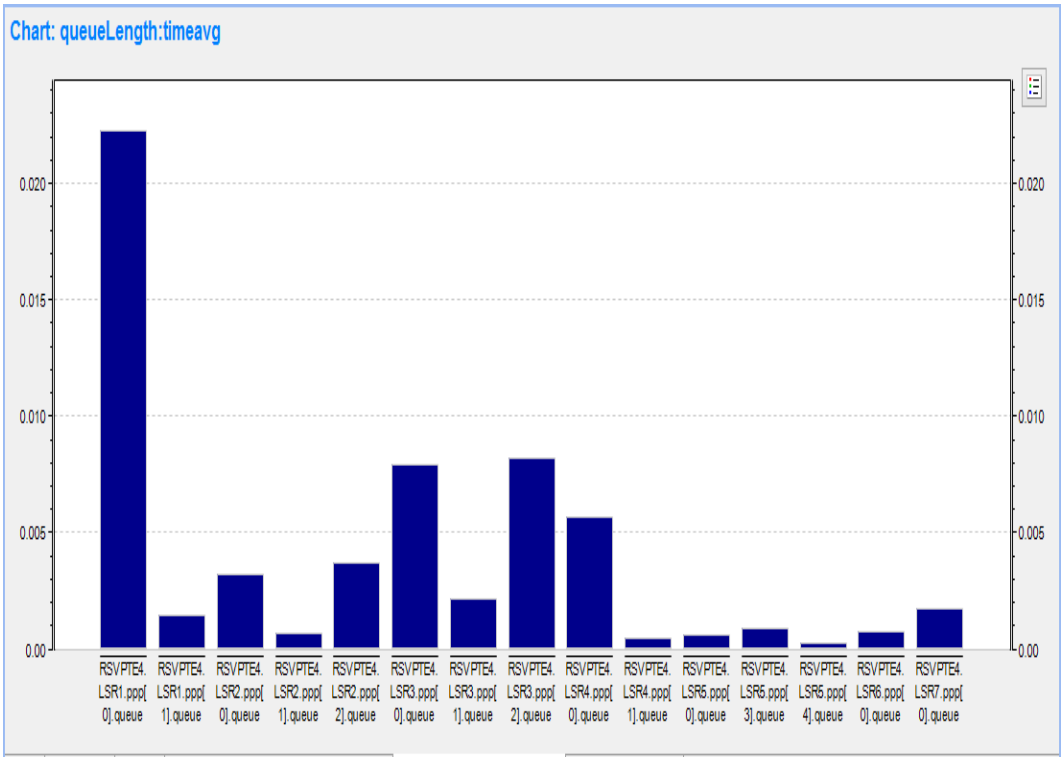
Packet loss non Optimasi	*Packet loss* Optimasi
78 %	0 %

3. Waktu Tunggu (*delay*)

   Waktu tunggu yang dihasilkan dari hasil pengujian simulasi menunjukkan adanya tingkat waktu tunggu (*delay*) yang relatif kecil dan mengalami penurunan peningkatan waktu tunggu ketika terjadi pengiriman suatu paket data pada model topologi jaringan optimasi yang menggunakan teknik *traffic engineering* MPLS. Hal ini dikarenakan adanya pembagian paket data sebelum dilakukannya pengiriman yang dilakukan oleh masing-masing LSR (*Label Switching Router*) pada model topologi jaringan optimasi dari hasil proses *broadcast information* yang dilakukan oleh masing-masing LSR (*Label Switching Router*). Hasil pengujian untuk mengetahui tingkat *delay* yang terjadi pada topologi jaringan sebelum dilakukan optimasi dapat di lihat pada Gambar IV-26. Hasil pengujian untuk mengetahui tingkat *delay* yang terjadi pada topologi jaringan setelah dilakukan optimasi dapat di lihat pada Gambar IV-27.



Gambar IV-26 Tingkat Delay pada Topologi non Optimasi



Gambar IV-27 Tingkat Delay pada Topologi Optimasi

Evaluasi pengujian selanjutnya akan di lakukan perbandingan hasil dari seluruh pengujian pada topologi sebelum dilakukan optimasi dengan topologi setelah dilakukan optimasi. Tabel IV-16 merupakan standar nilai *delay* yang sudah ditetapkan oleh *International Telecommunication Union* (ITU) [9].

Tabel IV-16 Standar Tingkat Delay ITU

Kategori	Tingkat *Delay*
Sangat Bagus	< 150 s
Bagus	150 s – 300 s
Cukup	300 s – 450 s
Kurang Bagus	> 450 s

Hasil yang didapatkan dari hasil pengujian pada topologi non optimasi maupun optimasi dengan melakukan pengujian pengiriman paket data suara (*voice*) pada varian interval waktu pengiriman 0.01 detik *delay* maksimum yang di dapat 0.00021ms atau sebesar $21x10^{-5}$ detik pada topologi non optimasi. Pada topologi optimasi di dapat 0.023ms atau sebesar $23x10^{-3}$ detik, tingkat delay pada

topologi optimasi mengalami puncak *delay* di awal ketik paket akan dikirimkan, hal ini disebabkan karena adanya proses LSR (*Label Swiching Router*) untuk membagi paket ke menjadi beberapa paket dengan ukuran terkecil yang disesuaikan dengan tingkat ukuran *volume bandwidth* pada jalu telekomunikasi.

Tabel IV-17 Perbandingan nilai Delay

Delay **Topologi Non Optimasi**	***Delay*** **Topologi Optimasi**
0.00021ms ($21x10^{-5}$ detik)	0.023ms ($23x10^{-3}$ detik)