Comparative Analysis of Packet Fragmentation on the MPLS Unicast IP Routing

Suraya Binti Zainuddin, Ruhani Ab Rahman
Faculty of Electrical Engineering
Universiti Teknologi MARA
40450 Shah Alam Selangor, MALAYSIA
surayaz@celcom.com.my, ruhani467@salam.uitm.edu.my

Abstract — **Multiprotocol Label Switching or MPLS is acknowledged and widely been used to overcome drawbacks introduce by traditional IP routing. This paper discussed on the network performance with the effect of packet fragmentation over IP and MPLS networks. In real implementation, fragmented IPv4 traffic causes a lot of problem such as increase load at router CPUs and also result in poor performance or even total communication failure. In addition, traffic fragmentation is used in numerous network attacks. Thus, we want to avoid the fragmentation at all or ensure the network is insulated from fragmented traffic. However, in some cases when using IPv4 fragmentation is unavoidable. Simulation models were developed using GNS3 to compare performance of Open Shortest Path First (OSPF) and MPLS network. Performance is determined by Round-Trip-Time (RTT), calculated throughput and packet loss. Analysis shows different protocols, data sizes and MTUs influence network performance. OSPF provides better RTT and throughput compared to MPLS with default MTU setting. However, better RTT and calculated throughput performance can be achieved by increasing the MTU for interface, IP and MPLS. Finally, the study also indicates packet fragmentation could degrade network performance.**

Keywords: **OSPF, MPLS, Unicast IP, Forwarding, LDP, GNS3, Shim Header, ICMP, Fragmentation.**

I. INTRODUCTION

Multiprotocol Label Switching (MPLS) is a standard architecture proposed by the Internet Engineering Task Force (IETF) that integrates label swapping forwarding with network layer routing. Over 300 Internet Drafts and numerous Requests For Comments (RFC) related to MPLS were produced and continues on refining the MPLS standards. This technology evolving in recent years and widely being implemented.

MPLS is a promising effort in order to deliver the traffic management and connection-oriented Quality of Service (QoS) support, speed up the packet-forwarding process, while retaining the flexibility of an IP-based network approach.

It also reduces the amount of per-packet processing required at each router in an IP-based network, which enhance router performance even more.

MPLS provides new capabilities in four areas that have ensured its popularity which are (i) QoS support, (ii) traffic engineering (TE), (iii) Virtual Private Networks (VPNs) and multiprotocol support .

Basically, MPLS overcomes problems found in conventional IP networks as well as the limitations of overlay models. Major drawbacks of traditional IP routing are [1]:

- All routers require routing protocol with full routing information.
- Routers only able to make a destination-based forwarding decision.
- Routers need to make a routing look-up for every single hop.

Multiple studies had been done on the performance analysis between MPLS protocol over conventional network [2][3][5]. MPLS provides better performance and easier traffic engineering (TE) compare to OSPF [2][3]. Packet drop behavior in MPLS is almost negligible amount compare to traditional IP network [3]. Besides, MPLS provides better throughput than conventional network [6].

Variety of tools are offered in the market for modeling and simulating MPLS networks such as GNS3, OpenSimMPLS and Opnet [4]. A study had been done on measuring MPLS overhead over Linux platform. The result reflected higher MPLS RTT compare to conventional IP [8].

Most of the research on comparing OSPF and MPLS focus on the traffic engineering [2][3][5][7] and virtual private network; which are the core applications in MPLS implementation. Very limited articles are written on the MPLS unicast IP routing performance. Thus, this paper mean to further explore on this basic MPLS concept.

This paper presented on the network performance using a network emulator known as GNS3. The performance is observed on how the fragmentation effects RTT, throughput and packet drops over OSPF and MPLS unicast IP forwarding. A detail study had been done on proving fragmentation lead to poor performance and reliability issue [17].

MPLS can be used for simple unicast forwarding which the packet forwarding logic is based on labels. During the selection of packet forwarding, MPLS considers only the routes available in the unicast IP routing table. Thus, the end result of using MPLS is equal to IP routing which have similar path forwarding. All other factors remain unchanged.

Basically, MPLS unicast IP forwarding does not offer any significant advantages by itself [9].

Unicast IP routing is the most common application for MPLS. Two mechanisms required on the control plane which are IP routing protocol and label distribution protocol [1]. Below figure illustrates simplified model of routing and forwarding mapping:

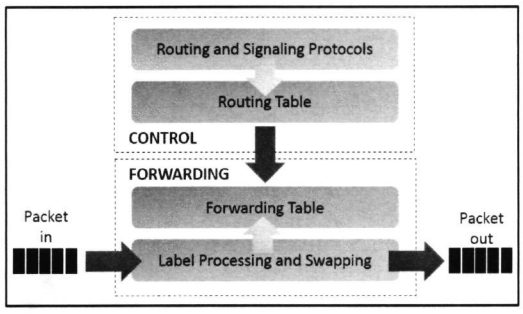

Figure 1: Mapping between routing and forwarding

II. Simulation Tools

Establishing and simulating the test environment for both OSPF and MPLS topologies for this study are using 3 tools; namely: GNS3 (Graphical Network Simulator), VMWare Player and Wireshark.

A. GNS3

GNS3 is a freeware graphical network simulator that allows users to design and deploy simulation of a complex network topologies at their ease. It is a complementary tool to real lab [10]. GNS3 all-in-one offers a package encompasses:

- Dynamips – the popular Cisco IOS emulator.
- VirtualBox – to run desktop and server operating systems as well as Juniper JunOS.
- Qemu – a generic open source machine emulator, its runs Cisco ASA, PIX and IPS.
- Wireshark – a packet capture freeware.
- Connection to virtual network/ host and real device.

Combination of these emulators provide complete and accurate simulation of real network [10]. In this study, network topologies are created using this software.

B. VMware Player

VMware player is a virtualization software which can run existing virtual appliances and create its own virtual machines. It is a free desktop application that allow user to run a virtual machine on a Windows or Linux PC [11]. This application allows:

- Virtual machine isolation
- Access to host PC devices
- Copy and paste between virtual machine and host
- Adjustable memory for optimal performance
- Powerful networking capabilities
- Configurable shutdown

C. Wireshark

It is a free and open-source packet analyzer. Wireshark is used for network troubleshooting, analysis, software and communication protocol development, and education. This freeware capable to understand the structure (encapsulation) of different networking protocols [12].

III. Simulation Model

The development of simulation models are based on the Telco Company C, but with simpler topology. The simulation model developed based on the following assumptions.

- All routers used in the topologies are Cisco C3640.
- All interfaces used in the topologies are serials with similar cost.
- One subnet (which consists of multiple routers (hops) in actual network) is represented by one router (one hop in test bed network).
- IP assignment is self-defined due to security purposes (not similar IP range as implemented in actual network).

Figure 2 illustrates the existing network topology for the site:

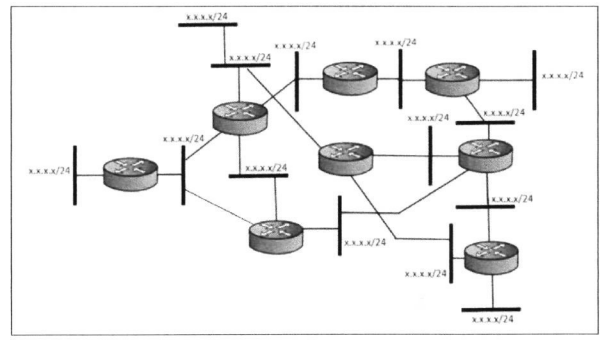

Figure 2: Actual Network Topology for the Selected Site

Actual network topology is simplified as per below figure:

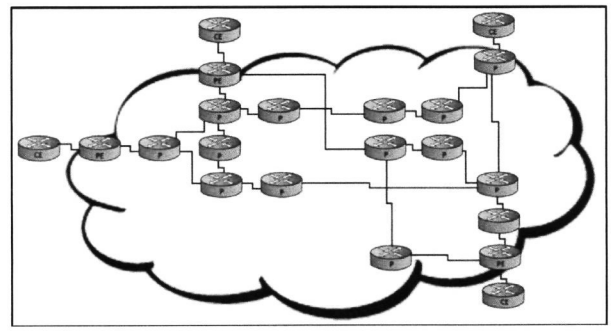

Figure 3: Test Bed Network Topology

Table 1 indicates the hardware technical configuration used for the test bed environment.

Table 1: Hardware Technical Configurations for Test Bed

No	Hardware	Configurations
1.	Host 1	**Processor:** Intel ® Core ™ 2 Duo CPU
		RAM: 128 MB
		Operating System: Microsoft Windows XP
		NIC: VMware Accelerated AMD PCNet Adapter
		Monitoring Tools: Wireshark Network Protocol Analyzer Version 1.10.10
2.	Host 2	**Processor:** Intel ® Core ™ 2 Duo CPU
		RAM: 2.5 GB
		Operating System: Microsoft Windows XP
		NIC: Broadcom Netlink ™ Fast Ethernet
		Monitoring Tools: Wireshark Network Protocol Analyzer Version 1.10.10
3.	Router M1 to M21	**Model:** Cisco 3640
		IOS: 3600 Software (C3640-JS-M), Version 12.4 (23)
		Fast Ethernet Interface: NM—1FE-TX
		Serial Interface: NM-4T
		Idle PC Value: 0x604d9334
4.	Channel Capacity	**Fast Ethernet**: 100 Mbps
		T1 Serial: 1.544 Mbps

Host 1 is connected to Router M1 and Host 2 is connected to Router M9. Host 1 is connected to the physical network card on the host machine that run GNS3. While Host 2 is connected to virtual machine that run on the similar machine. Each host is furnished with Wireshark, a network protocol analyzer.

The test environment comprises 21 routers inclusive of 4 Customer Edge (CE) routers, 4 Provider Edge (PE) routers and 13 Provider (P) routers; as shown in Figure 3. Cisco C3640 routers are tuned to the optimized idle PC value in order to obtained a stable network topology on GNS3. ICMP network protocol packets are sent from Host 1 to Host 2 via command prompt on the host machine to observe the network performance.

OSPF routing configured on all routers in order to setup OSPF routing based network. IP routing protocol function is to carry the information regarding the reachability of networks [1]. The following is a sample of command for OSPF routing configuration:

```
router ospf [process-id]
 network [ip address] [mask] area [area-id]
```

In this experiment, LDP is selected for label distribution protocol for label binding over network learned via the routing protocol [1]. MPLS is enabled on the router's interfaces to establish MPLS unicast IP forwarding as following:

```
interface [type-number]
 mpls ip
 mpls label protocol ldp
```

The following sample of command is issued to change router's interfaces and IP MTU :

```
interface [type-number]
 mtu [value]
 ip mtu [value]
```

MTU size set to 1512 to cater additional 3 labels of 4 byte for MPLS labelling. Below basic configuration is configured on the router's interfaces to allow MPLS MTU size to be changes to value required:

```
interface [type-number]
 mpls mtu [value]
```

Above all mentioned command, "ip cef" needs to be enabled which by default it is already turns on for Cisco router C3640.

Traceroute is done to check on the path established for packet travelling from Host 1 to Host 2. Below screen shot shows traceroute from Host 1 with OSPF routing established:

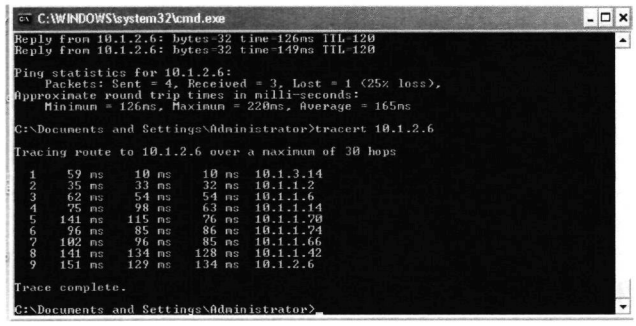

Figure 4: Screenshot of Traceroute from Host 1 to Host 2 (OSPF Routing)

By issuing traceroute at the router, MPLS labelling can be seen on the forwarding path once MPLS was enabled as per Figure 5.

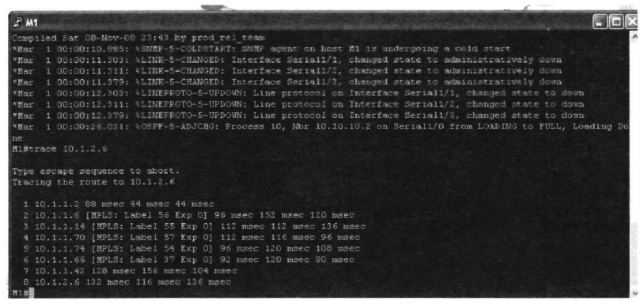

Figure 5: Screenshot of Traceroute with MPLS implementation

MPLS labels and stacking bit are observed from the experiment at every hop in the packet's routing path from Host 1 to Host 2 using Wireshark and Cisco commands.

Figure 6 depicts the label swapping flow and stack bit monitored for MPLS unicast IP forwarding during the experiment. A label is assigned to every destination network in the IP forwarding table and stack bit is set to 1 to indicate single label with 32 bits inserted between Layer 2 and 3 for the MPLS frame mode [1].

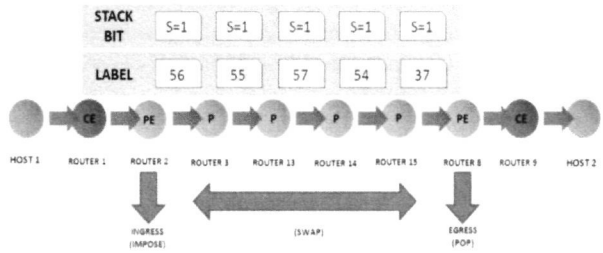

Figure 6: MPLS Labels and Stack Bit

Figure 7 illustrates test bed topology established in the GNS3:

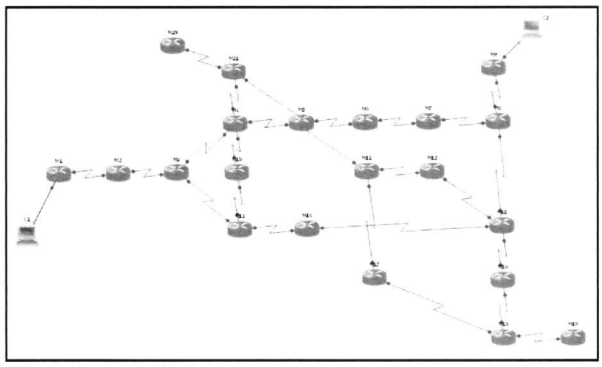

Figure 7: Test Bed Network Topology in GNS3

During the experiment, packets are sent with and without Don't Fragment (DF) bit. It is to determine the maximum transmission unit (MTU) size and observe the point of fragmentation to happen on the network path between the 2 hosts.

The entire test is done systematically to ensure the stable data readings. The experiment done as per Figure 8 flow:

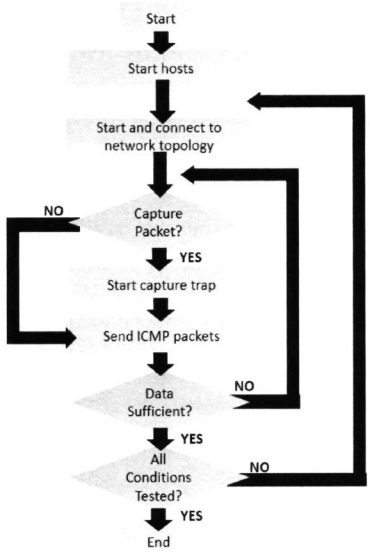

Figure 8: Test Flow

IV. RESULTS

This section presents several results generated from the simulation using a network emulator. ICMP packets were issued using Ping command to obtain RTT between 2 hosts. Throughput is calculated based on the RTT and packet loss was observed.

A. Variation of Packet Size in OSPF and MPLS Topologies

Figure 9 and Figure 10 show average RTT for both OSPF and MPLS unicast IP forwarding with default MTU of 1500. ICMP packets size are varied to 10, 50, 100, 500, 1000, 1500 and 2000 bytes.

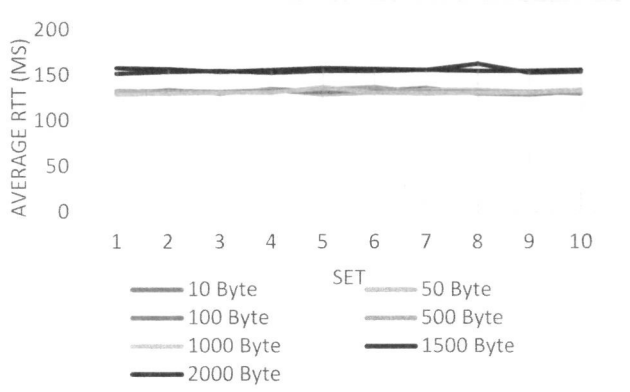

Figure 9: OSPF RTT Performance without DF Bit Set

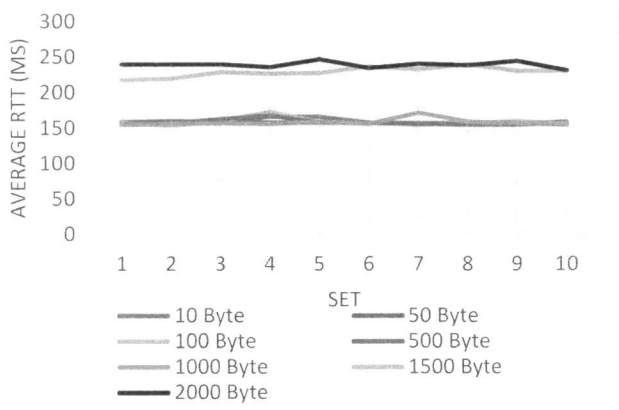

Figure 10: MPLS RTT Performance without DF Bit Set

RTT resulted almost similar readings for OSPF and MPLS with DF bit sets. Unfortunately, at 1500 and 2000 bytes size sent, host received ICMP error of "Packet needs to be fragmented but DF set" for both topologies.

By using the RTT readings, a theoretical throughput can be calculated using below equation [14]:

$$TCP\ Throughput = \frac{TCP\ Window\ Size\ (bits)}{Round\ Trip\ Time\ (s)}$$

Equation 1: Maximum TCP Throughput

However, this equation does not cater packet loss condition. Default window size for Windows XP Operating System is 17,520 bytes which is equivalent to 140,160 bits [13][14][15] which produced below values.

Table 2. Maximum Theoretical TCP Throughput on Windows XP

Test Condition	DF Set	Ave.RTT (ms)	Throughput (Kbps)
OSPF			
Packet Size < 1500 bytes	No	132	1.061
	Yes	132	1.061
Packet Size ≥ 1500 byes	No	155	0.904
	Yes	Packet needs to be fragmented	
MPLS			
Packet Size < 1500 bytes	No	158	0.887
	Yes	158	0.887
Packet Size ≥ 1500 byes	No	234	0.599
	Yes	Packet needs to be fragmented	

Figure 11 visualizes the packet drop behavior in OSPF network without DF bit set. Occurrence of packet drop is more frequent for packet size larger or equal to 1500 bytes.

While, Figure 12 illustrates the packet drop in OSPF network when DF bit set. Packet drop is almost negligible.

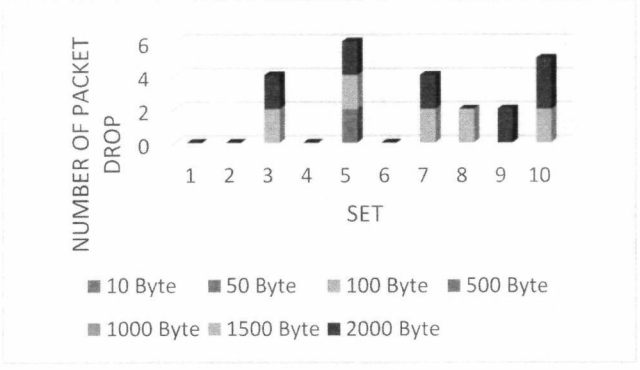

Figure 11: Number of Packet Drop for OSPF without DF Bit Set

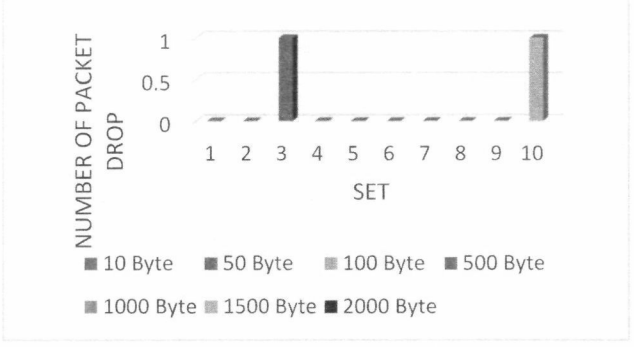

Figure 12: Number of Packet Drop for OSPF with DF Bit Set

Number of packet drop in MPLS topology without DF set is pictured as per Figure 13. Similar as per OSPF, packet loss is frequent for packet size larger or equal to 1500 bytes However, no packet drop is observed when DF was set.

Figure 13: Number of Packet Drop for MPLS without DF Bit Set

B. Variation of MTU in OSPF and MPLS Topologies

In this section, interface and IP MTU are varied from default of 1500 to 1512 and 1600 for OSPF. While for MPLS topology, interface, IP and MPLS MTU are varied from with similar setting as per OSPF. Changes in MTUs are done to the PE and P routers.

Figure 14 shows average RTT when ICMP packets send with fragmentation allowed for default MTU = 1500.

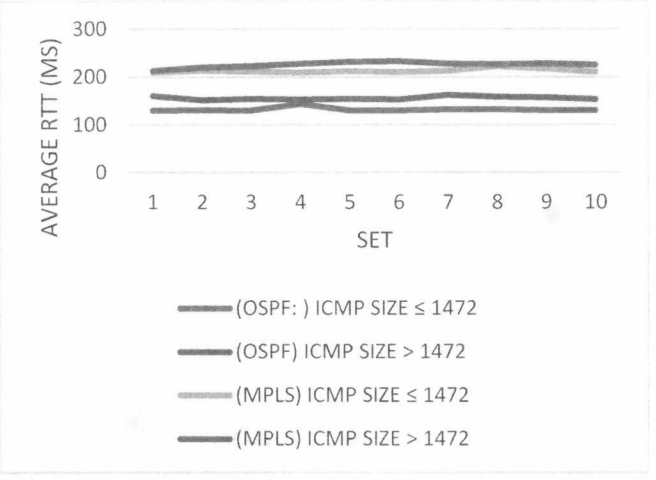

Figure 14: Average RTT without DF Bit Set (MTU = 1500)

Figure 15 presents average RTT when ICMP packets send with fragmentation allowed for default MTU = 1512.

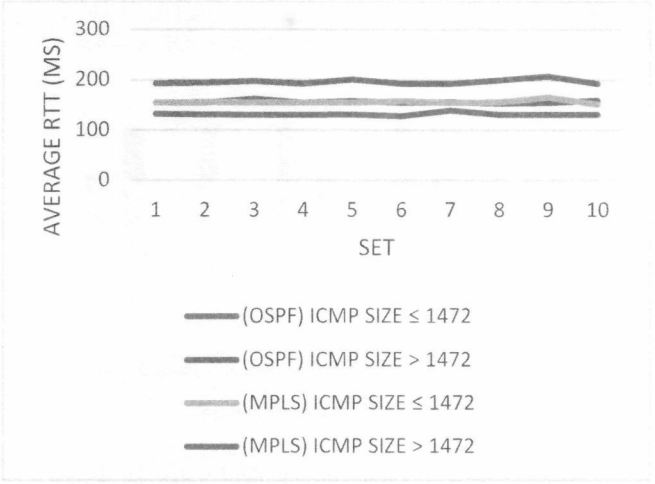

Figure 15: Average RTT without DF Bit Set (MTU = 1512)

Figure 16 indicates average RTT when ICMP packets send with fragmentation allowed for default MTU = 1600.

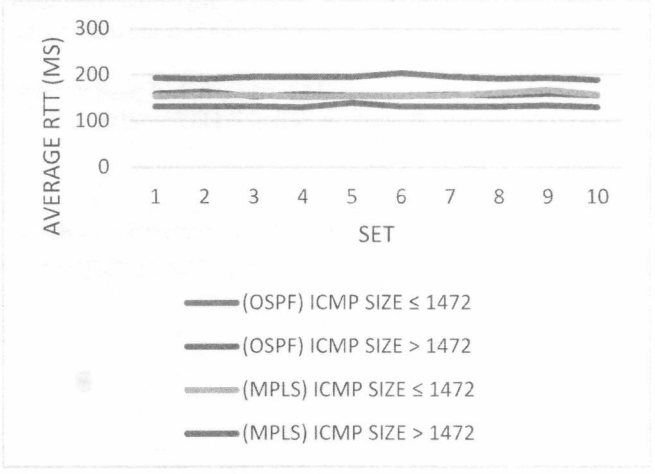

Figure 16: Average RTT without DF Bit Set (MTU = 1600)

Below Figure 17 illustrates the average RTT for both OSPF and MPLS networks when DF bit is set. When packets sent with DF, no fragmentation was allowed.

Again, maximum theoretical TCP throughput is calculated using Equation 1. The results are tabulated in Table 3.

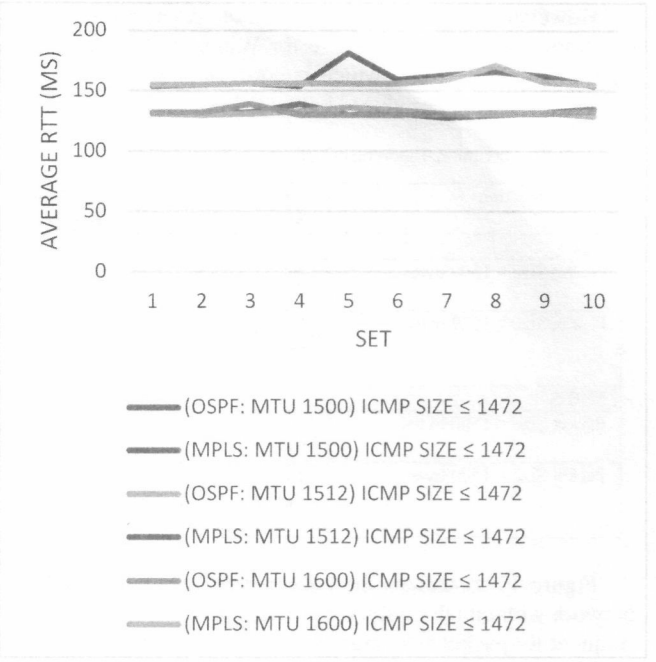

Figure 17: Average RTT with DF Bit Set

Table 3. Maximum Theoretical TCP Throughput on Windows XP

Test Condition	MTU	ICMP Size > 1472 Byte	Ave.RTT (ms)	Throughput (Kbps)
OSPF				
Fragment	1500	No	132	1.061
		Yes	156	0.898
	1512	No	131	1.069
		Yes	156	0.898
	1600	No	132	1.061
		Yes	157	0.893
Don't Fragment	1500	No	132	1.061
		Yes	Packet needs to be fragmented	
	1512	No	132	1.061
		Yes	Packet needs to be fragmented	
	1600	No	132	1.061
		Yes	Packet needs to be fragmented	
MPLS				
Fragment	1500	No	213	0.658
		Yes	225	0.623
	1512	No	156	0.898
		Yes	196	0.715
	1600	No	156	0.898
		Yes	194	0.722
Don't Fragment	1500	No	Packet needs to be fragmented	
		Yes	Packet needs to be fragmented	
	1512	No	160	0.876
		Yes	Packet needs to be fragmented	
	1600	No	158	0.887
		Yes	Packet needs to be fragmented	

Number of packet drop in both networks without DF set for default MTU is pictured as per Figure 18.

Figure 18: Number of Packet Drop for OSPF and MPLS without DF Bit Set (MTU = 1500)

Figure 19 displays number of packet drop in OSPF and MPLS without DF set for MTU 1512. While, number of packet drop in both topologies without DF set for MTU 1600 is depicted as per Figure 20.

Figure 19: Number of Packet Drop for OSPF and MPL without DF Bit Set (MTU = 1512)

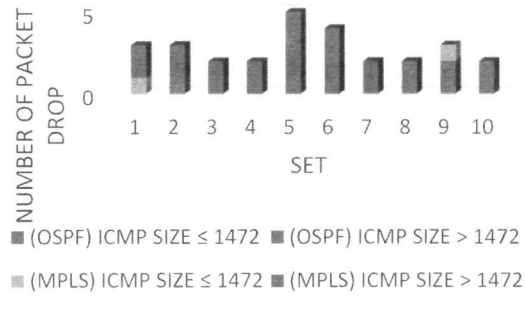

Figure 20: Number of Packet Drop for OSPF and MPLS without DF Bit Set (MTU = 1600)

No packet loss encountered in both networks with MTU varies from1500 to 1512, and 1600 when ICMP packets are sent with Don't Fragment bit.

V. Discussions and Results

Data obtained is observed and analyzed based on ICMP RTT, calculated throughput and packet loss. Figure 9, Figure 10 and Table 1 in the previous section show that incremental in packet size doesn't have significant impact on the RTT and throughput; as long as packet size is smaller than the MTU and no fragmentation occurs in both topologies. Packet drop is negligible.

However, once the packet size increases more than the MTU; fragmentation will happen. RTT increases and it decreases the calculated throughput. Occurrence of packet drop is frequent. Half or more of the data captured from the runs perceive to have packet loss around 0.0001%. This can be observed from Figure 11 and Figure 13.

Similar RTT response for both topologies noticed when packet send without fragmentation allowed (DF is set) using default MTU. However, once the packet size reached 1473 for OSPF, it will be dropped. This is due to ICMP packet send with the addition of 28 bytes of IP header resulted size more than 1500 (default MTU value). While, similar response discovered when MPLS unicast IP packet achieved 1469 bytes. In the MPLS network, 4 bytes lesser of ICMP packet size can be sent compare to OSPF caused by the allocation for 32 bits MPLS shim header.

As the continuity, test had been done by varying the MTU value and ICMP data packets are set at size 1472 and 1473 bytes. Similarly, RTT is higher in both OSPF and MPLS networks once packets are fragmented and apparently, MPLS RTT will be higher than OSPF due to label overhead processing. This is concluded in Table 3.

Almost stable data readings are obtained for OSPF when MTU was changed from 1500 to 1512 and 1600. However, different RTT performance observed for MPLS as per Figure 14 and Figure 15. When MTU is change from to 1500 to 1512, there is a remarkable improvement in the RTT performance which relates back to the calculated throughput. This is due to packet was not fragmented when MTU increased to 1512. While during MTU set to default, ICMP Packet of 1473 is split into 2 (1472 bytes and 1 byte).

Thus, the exact response was expected when MTU set to 1600 as per Figure 16. It resulted the almost similar reading as MTU 1512. MTU was increased at the provider-edge and provider routers. However, MTU remained 1500 at the customer-edge points. This is the reason for packet to still being fragmented at 1500 bytes even though core network router's MTU increased. It instantly set the size limitation for packet to be sent without fragmentation.

The packet drop response is homogeneous. Persistent packet loss around 0.0001% discovered when packet was fragmented. This packet drop responses are visualized in Figure 18, Figure 19 and Figure 20.

Overall, OSPF performance is better than MPLS with unicast IP routing in term of RTT and throughput. RTT for MPLS seems to be slightly higher due to the introduction of label to each packet send. In this case, 4 bytes label is appended to each packet send out with MPLS applied. Packet loss behavior is similar in both network which more visible

when fragmentation happened. The effect of label stacking to performance is studied to cause higher RTT [8]. RFC4963 and a study had been made on proving that 16-bit IP identification field is not enough to prevent frequent incorrectly assembled due to fragmentation in IPv4 [16][17].

VI. Conclusion

This paper analyzed performance of OSPF and MPLS unicast IP routing topologies based on packet fragmentation. Packet size and MTU are chosen as variable in proposed simulation model. Several scenarios were configured and tested using Cisco C3640 routers with Windows XP environment hosts. Results obtained have been compared in terms of RTT, calculated throughput and packet loss.

Obviously, OSPF has better performance compare to MPLS either packet is fragmented or vice versa. As earlier iterated, MPLS unicast IP forwarding itself does not offer any benefit. However, when it comes to MPLS competent applications such as TE and VPN, MPLS unicast IP routing is a compulsory. Thus, in the case of MPLS unicast IP routing to run by itself without other applications, OSPF is suggested and preferred due to better performance of RTT and throughput measurements. It is suggested to avoid fragmentation by sending small datagram or discover minimum MTU of the path [17].

However, this study does not look into detail on how MPLS unicast IP routing provides advantage in terms of IP looping prevention. This is the capability that can be compromised for the performance degradation compare to OSPF.

Future study can be done on Transport Control Protocol (TCP) and User Datagram Protocol (UDP) throughput observation together with IP looping prevention in MPLS Unicast IP routing.

Acknowledgment

I would like to express my gratitude to Associate Professor Ruhani Ab Rahman, for her guidance throughout the project and thanks to all authors from whom I obtained all the information for this study through their writings, documentations and slide presentations.

References

[1] Cisco Systems, "Implementing Cisco MPLS v2.1", 2002.

[2] Thomas Kramer, Prof. Anja Feldmann, "IP Traffic Engineering – OSPF versus MPLS", Computer Science Department, University of Saarland, 2003.

[3] Md. Arifur Rahman, A.H. Kabir, K.A.M. Lutfullah Z. Hassan, M.R. Amin, "Performance Analysis of MPLS Protocols over Conventional Network", Department of Electronics and Communication Engineering, East West University, 2008.

[4] Azeddien M. Sllame, "Modeling and Simulating MPLS Networks", Faculty of Information Technology, University of Tripoli, 2014.

[5] Stefan Kohler, Andreas Binzenhofer, "MPLS Traffic Engineering in OSPF Network – a Combined Approach", Institute of Computer Science, University of Wurzgburg, 2003.

[6] Mahesh Kr. Porwal, Anjulata Yadav, S.V. Charhate, "Multimedia Traffic Analysis of MPLS and Non MPLS Network", International Journal of Computer Science and Applications Vol. 1, 2008.

[7] Mahesh Kr. Porwal, Anjulata Yadav, S.V. Charhate, "Traffic Analysis of MPLS and Non MPLS Network including MPLS Signaling Protocols and Traffic Distribution in OSPF and MPLS", First International Conference on Emerging Trends In Engineering and Technology, 2008.

[8] A. Pescape, S.P. Romano, M.Esposito, S. Avallone, G.Ventre, "Measuring MPLS Overhead", Department of Computer and Systems, University Federico II of Naples, 2008.

[9] Wendell Odom, Rus Healy, Denise Donohue, "CCIE Routing and Switching Certification Guide", Cisco Press, 2009.

[10] GNS3, "http://en.wikipedia.org/wiki/GNS3"

[11] VMware Player, "http://en.wikipedia.org/wiki/VMware_Player"

[12] Wireshark, "http://en.wikipedia.org/wiki/Wireshark"

[13] Elliotte Rusty Harold, "Java Networking Programming", O'Reily Media, 2013.

[14] Why you maximum throughput less than your bandwidth, "http://networksolutionexperts.com/why-your-maximum-thru-put-is-less-than-your-bandwidth/"

[15] Walter Goralski, "The Illustrated Network: How TCP/IP Works in a Modern Network", Morgan Kaufmann, 2009.

[16] RFC 4963, "http://www.isi.edu/in-notes/rfc4963.txt"

[17] Christopher A. Kent, Jeffey C. Mogul, "Fragmentation Considered Harmful", Western Research Laboratory, 1987.