Indian Journal of Science and Technology, *Vol 9(28), DOI: 10.17485/ijst/2016/v9i28/98081, July 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Comparative Analysis of TCP/IP and ROACM Protocols- A Simulation Study

Abdullah Ali Bahattab*

College of Telecommunications and Electronics, Computer Technology Department, Jeddah, Saudi Arabia; abahattab@gmail.com

Abstract

The purpose of this research is to introduce a new novel protocol, Route Once and Cross-connect Many (ROACM), and compare it against the Transmission Control Protocol/Internet Protocol (TCP/IP). The Open Shortest Path First (OSPF) is one of the interior routing protocols, which creates the routing tables for routers. These tables are adopted by TCP/IP protocol for forwarding packets. In this research, the routing protocol used to build the tables in routers is OSPF, but the comparison is between the TCP/IP and ROACM protocols. The variables selected for studying the efficiency of TCP/IP and developed protocol are average delay and throughput. The developed protocol outperforms in both the average delay and the throughput. Also, it was found out that developed protocol namely ROACM protocol is exponentially more efficient than TCP/IP.

Keywords: Delay Time, Route Once and Cross-connect Many, Routing, Routing Protocol, Swtiching, TCP/IP, Throughput

1. Introduction

The increased traffic transferred through ISPs (Internet Service Providers) caused by new access methods such as ADSL[1], last mile fiber optic, and wireless connections is a burden on routers, especially backbone routers. While traveling from source to destination, most of the IP packets encounter different networking devices which have different network semantics (Network Interfaces [NIs]) such as packet forwarding routers, frame relay, MPLS, ATM, etc. The ROACM provides many features to intelligent routers/switches. In this work, our novel protocol, ROACM, has been explained where the IP packet contains an extra header that allows a dynamic virtual circuit to be created. This header contains all relevant information to cross-connect the IP packets at the second layer (Data Link). In the call set up stage, the information is attached to the network layer header while in data transmission stage, the information is stored in the ROACM header (in the frame)[18]. Propagation of the ROACM information can occur below the IP level in networks that contain routers that all agree and support ROACM. Since this information is appended to the end of an IP packet, the routers, which do not employ ROACM are still able to forward packets using regular routing protocol. The ROACM itself maps virtual circuit links (indexes), which are provided from the local interface tables at each router where each index corresponds to a next hope interface address. This allows interoperability on a wide range of networks[3]. This novel protocol has many advantages. The IP packet does not reach the network layer except in the call set up stage. Also, there is no kind of search or routing/switching protocol distribution needed since all information is already encapsulated in the ROACM header at layer two[18]. Moreover, it is easy to implement. This means no need for high power processing in the routers. Furthermore, packets reordering in the Internet can be eliminated. In this paper, we show how the ROACM protocol is exponentially more efficient than TCP/IP using the network simulation tool NS3.

This paper is organized as follows: In Section II, we explain in details the ROACM protocol. Section III, TCP/IP is briefly introduced. Section IV focuses on the performance analysis. Also, the findings of the research are discussed. Then our conclusion in Section V.

**Author for correspondence*

The techniques for building fast forwarding packets routers/switches are implemented in different ways. Different mechanisms for a router is required for operation at high-speed. Various techniques adopted for routing/switching like architecture of cell switching router[4], MPLS (multiprotocol label switching)[5–10], internet protocol switching[11] and tag switching[12,13] that are effort for standardization underway at the task force of internet engineering. Tag switching depends on fundamental components such as control and forwarding. The forwarding component adopts the information about tag taken by protocols and the information about tag-forwarding is controlled by a switch of tag for forwarding the packet. The component of control is suitable to maintain correct information about tag-forwarding among a cluster of tag switches which is interconnected[13]. MPLS is the framework for the protocol on which the focus on the provider of network service is concentrated as it gives unbreakable and privacy security to the clients. Therefore, it is adopted by backbone network of ISPs for forming the architecture of MPLS tunneling like MPLS virtual private network. ISPs must be kept with time by changing to new technologies which is the backbone for maximizing the share of market. MPLS is developing technology and it is the best solution to present internet protocol network issues. It gives more capability for traffic engineering when compared with the other counterparts. MPLS performs in combining with routing of internet protocol and its major aim is to give the switching speed to 3rd layer. Introducing the labels gives the effective substitute and evades the requirement of huge table for routing lookups and outcomes in speed routing. At the same time MPLS' telling factor has its capability to control and categorize the traffic for providing better resources utilization. Such technology is adopted for resolving integration and issues of traffic engineering in carrier networks. MPLS virtual private network gives advantages that providers of service require in their network like reliability, scalability and manageability[14]. MPLS fasten the forwarding of packets speed on the basis on labels and minimizes the adoption of table for routing look up from routers to routers of label edge while routers of label switch adopt protocol of resource reservation or label distribution protocol for allocation of label and table of label for forwarding the packet. In [15] dynamic protocol is deployed that handles updates packets for label switch paths details associated with mechanism for feedback is also deployed which identify shortest path among the network of MPLS and feedback is given for overcoming congestion, such mechanism was based on hop by hop instead of end to end thus giving a faster and reliable and path of congestion free for packets. Chan et al.[16] studied about high performance internet protocol forwarding with effective update of routing table. To enhance BSP (bulk synchronous parallel) for prefixes this is more than 16 bits with an algorithm of fast-forwarding table. The update cost links with the process of construction of table, this research addressed the incremental progress for computing already and developed a fast forwarding-table algorithm for construction. With the changed tree of multi-way search, it minimizes the tree depth and removes the pointers storage. This would result in minimizing the size of FT (Fault Torrent) and shortens the look up time of routing table. When focusing on the route flaps, the performance of forwarding will minimize by only 3.1 percent with updates of 4000 BGP per 30 second in the worst case. In the ROACM, no need to search in the routing table except for the call set up packet. Also, no table manipulation is needed since the ROACM depends on the current routing table.

2. Description of the Roacm Protocol

The proposed protocol consists of four major tasks. First; call set up. Second; data transmission. Third; path update, and finally; recovery plan. In this protocol, the IP packet will have an extra header such that it has the static and dynamic fields, shown in Figure 1. The static (control)

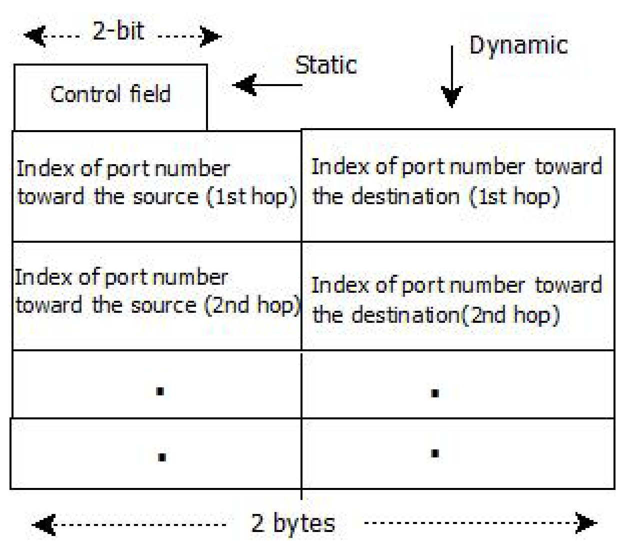

Figure 1. The IP packet's extra header.

field is used to establish connection/ ACK connection / data transmission (2-bit), where '01' means establish connection, '10' means ACK connection, and '11' means data transmission. The dynamic fields are the following pairs of two bytes in the header that contain the index port numbers for the source and the destination NI addresses (next hop) one byte for each direction. In other words, each pair of these two bytes represents the information of the outgoing ports (NI addresses) for next hop (a router) for both directions. The more hops from a source to a destination, the more pairs of the two bytes are added to the header. This addition is only performed in the call up stage with maximum of 255 hops. This Maximum number can be changed if needed. In a data transmission stage, only one column in Figure 1 plus the control field are used according to the direction of the IP packet. Also, each router, in an IP packet's path, should have its local router interfaces table, which is stored locally. In Figure 2, the table has two fields. The index field and the NI address field. Thus, all the interfaces in a particular router will be listed in this local router interfaces table. This index's value will be stored in one byte of the pairs of the two bytes in the header as shown in Figure 1. This is to avoid storing the NI address (for example MAC address is 48-bit) in the header. Having all this information in a ROACM header (in an IP packet) and the local router

interfaces table at each router, a router will immediately check the relevant fields in both the ROACM header and the local router interfaces table and shunt the IP packet to the next hop. Thus, all IP packets going to the destination, routers don't need to check the IP address for each packet any more, but just pick up the index port number field located in the ROACM header, which is attached to all IP packets and immediately get the NI address corresponding to that index from the local router interfaces table and use the NI address to cross-connect all IP packets. Moreover, packet reordering in the Internet can be eliminated. This can be achieved by using the index in the header to associate it with a complete path through the router. This way packet reordering inside the router can be eliminated, thereby increasing the speed and reducing the delay in the network. Therefore, routers, using this protocol, can cope with the enormously increased traffic

Index	NI address
1	NI1
2	NI2
3	NI3

Figure 2. Local Router interfaces table.

2.1 Call Set Up

At the set up stage, the source workstation sets the control field (establish connection/ ACK connection /data transmission) with value '01', which means establishing the connection. Therefore, a first regular IP packet will be sent to establish the connection and collect the ROACM information. Also, each router should have a local router interfaces table, which contains an indexed list that each index corresponds to an interface (NI address) connected to the router (this list is stored locally in each router) as shown in Figure 2. When the IP packet reaches an edge router, this router searches for the destination IP address in the forwarding (or routing) table. The main aim of this search is to find the corresponding NI address (Output Interface) for the next hop (router). Many times, edge routers have default route. Once it finds it, the router will use this information to forward the IP packet. But firstly, the router adds this index port number in the ROACM header in the IP packet specifically, in the right side byte of the first pair since it is the first hop toward the destination, see Figure 1. These indexes correspond to the NI addresses in the local router interfaces table (for example, the index value '13' instead of storing the NI address in the header, which in case of a MAC address, the NI consists of 48 bits), Immediately after finishing the above process, the router gets the NI address for the opposite direction (Input Interface). It can be found in the destination physical address at the data link layer of the same packet. Then the router stores the corresponding index port number in the IP packet's extra header, specifically in the left side byte of the first pair since it is the first hop from the source, see Figure 1. Therefore, this pair of information (the two bytes) belongs to this particular router. Notice that this information is attached to the IP packet's extra header in a call set up while half-size of this information will be stored to the ROACM header in ACKconnection and data transmission stages. Then the IP packet will move to the next hop (enterprise or core router). At this router (the second hop), the router will repeat the same process again as the first router did. That is, the router finds the output NI address by searching the routing (or forwarding) table, and finds the Input NI address by getting it directly from the same incoming packet and gets their corresponding index port numbers from the local router interface table. After that, the router will add these index port numbers in the IP packet's extra header in the second pair of the two bytes. Finally, the router forwards the IP packet to the outgoing interface. This process will continue until the IP packet reaches the destination (edge) router. Therefore, each pair of the two bytes represents the indexes of the corresponding network interfaces for a particular router in the IP packet's path. One side of these bytes for forward direction while the other for the backward direction. In other words, the right side column as shown in Figure 1 is for the forwarding direction while the left side is for the backward direction. All information in Figure 1 is stored in the source and the destination workstations. When the IP packet goes in the forward direction, only the forward information and the control field are attached, see Figure 3. And when the IP packet goes in a backward direction, only the backward information and the control field are attached. Hence, half-size of the information iscarried in the ROACM header. Thus, in the data transmission stage, the router will immediately look at this information to get the index numbers and find the corresponding NI addresses stored locally in each router in the IP packet's path to move to a next hop. So, it forwards the IP packet without looking up in the cache/forwarding (or routing) table. The source and the destination workstations are responsible to add the right information (forward or backward header) in the ROACM header because the workstations (source and destination) keep a copy of the information as in Figure 1, where the ROACM header encapsulation starts. Also, the workstations (source and destination) keep a copy of their edge routers' MAC addresses, so that it can send the IP packet immediately.

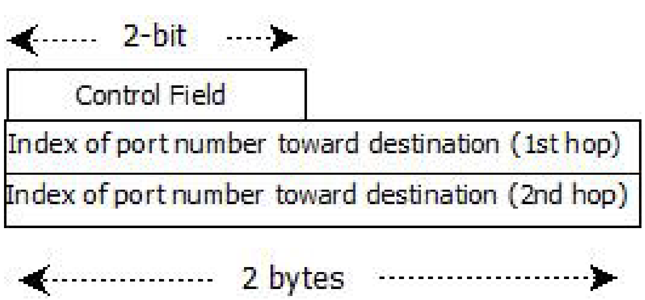

Figure 3. The ROACM Forward Header (to the destination).

By the time the first IP packet reaches the destination workstation, this IP packet will have all information about the index port numbers, which their NI addresses are stored locally in each router (hop) in the IP packet's path. This information is in the forward and the backward header. A copy of this information is kept in the destination workstation. The acknowledgment will be sent back to establish the connection (as specified by TCP/IP protocol), where a copy of the complete information is attached as a data to be stored in the source workstation. Before sending the IP packet back, the destination workstation sets the establish connection/ ACK connection /data transmission control field with value '10', which means the IP packet is coming back with the header information as a data. Also, it uses only the backward header information in the ROACM header. Then, the destination workstation sends back the IP packet to the source workstation that contains the complete header information and the connection ACK. Therefore, the IP packet will be cross-connecting from the destination workstation to the source workstation.

2.2 Data Transmission

Once the IP packet reaches the source workstation with value of '10' (ACK the connection), it knows that the data of this IP packet is in fact the complete header information, and the connection is established. A copy of this information should be stored in the source workstation. The source workstation will attach a copy of only the forward ROACM header information in each subsequent IP packet that it sends out to the destination with the control filed of value '11', which means data transmission. For all subsequent IP packets going to the destination workstation, routers do not need to check the IP addresses anymore nor they perform switching, but only look at the index port number field in the ROACM header at each packet and immediately get the corresponding NI address, stored in the local router interfaces table and use it to cross-connect the IP packets. Thus, there is no routing or switching, but only cross-connecting.

2.3 Path Update

After establishing the connection, data will be transferred forth and back between the source and the destination workstations. It is not necessary that this route will continue to be the optimal path. In other words, when establishing the connection, it could be at time=t0 was the optimal route that was stored in the ROACM header but it is not necessary that the same is true at time=t1. Thus, a periodic refresh message should be sent to update the optimal route.

2.4 Recovery Plan

If any output port malfunctions, the router should send a message to the source workstation to stop sending any more IP packets and should re-establish the connection to start new session.

2.5 Performing Cross-conncting

The header information is stored in the ROACM header in a stack fashion (LIFO). Also, the control filed (2-bit) will be attached at the top of the stack. In Figure 4, if a connection is established, as explained in section III. A, with three routers (hops), then we would have 6 bytes of header information (three for forward header information, right column with the values '4', '6', '8', and the other three for the backward, left column with the values '1', '3', '5') plus 2-bit control field header with the value '10', see Figure 5. The destination work station, sends back the IP packet, which contains the header information as data. It encapsulates only the backward information and the control field in the ROACM header such that the first hop (1-byte) information (as shown in Figure 5) is located at the last position in the stack and the last hop is at the top of the stack and then the 2-bit control filed, see Figure 6a.

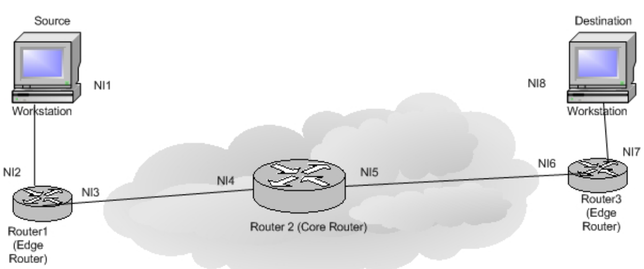

Figure 4. Example of a network with three hops.

10	
1	4
3	6
5	8

Figure 5. The information header for the network in Figure 4.

In the Data Link Header (DLH), the Source MAC address (belongs to the workstation) and the Destination MAC address (belongs to the Ethernet router interface) are stored in the workstation for fast transmission, since the workstation knows its NI and also can have a copy of the router NI. Once the IP packet reaches router 3, it extracts and checks the 2-bit control filed. If it has the value '10' or '11', it performs cross-connection. Since it is an ACK connection, the value must be '10'. Then, router 3 extracts the first byte (top of the stack), which has the index number of the NI address stored in the router. In this example, the value is '5'. Then, the router checks in the local router interfaces table for corresponding NI address towards the source workstation. This byte of header information will be discarded. Then router 3 encapsulates the new frame with the new (shrink) ROACM header, where one byte is dropped (the value '5'), and re-add only the 2-bit control field with same value '10', see Figure 6b. Therefore, we have, in the new ROACM header, only two bytes and again at the top of the stack is 2-bit control field. In the next hop, router 2 extracts and checks the first two bits and then the first byte and gets the corresponding NI address stored locally in the router. Again, router 2 will discard this byte of header information (the value '3') after finding the corresponding NI address, and adding back the 2-bit control

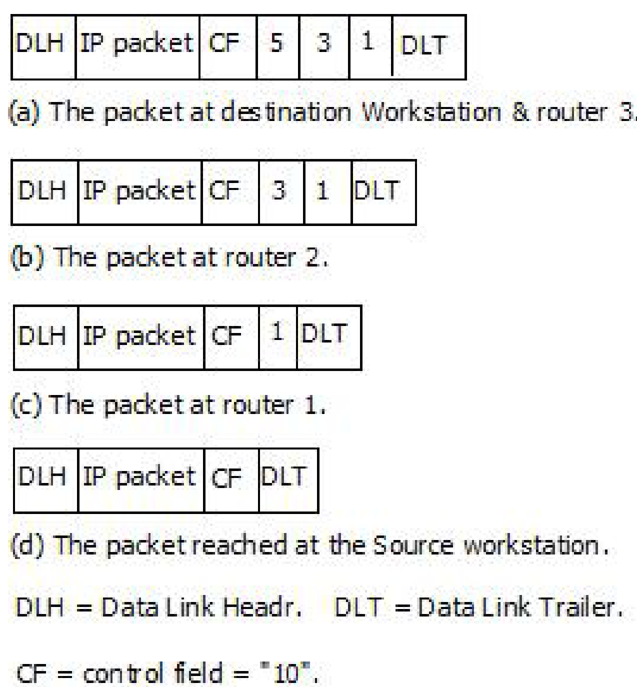

Figure 6. The modification on ROACM Header when moving from a destination to a source workstation.

field with same value at the top of the stack. After the new ROACM header is encapsulated with the current destination NI address, it will be sent for the next hop. In this case, only 2-bit and one byte are left for the last hop, see Figure 6c. The remaining byte is the index number for the NI address for the source workstation. At the last hop, router 1 does the same process that it extracts and checks the control field and the remaining one-byte header. Then, router1 gets the current NI address, discards the byte of header (the value '1', but will keep the control bit), and re-encapsulates the IP packet, see Figure 6d. Once the IP packet reaches the source workstation, it will know that this is the ACK for the connection, and stores the information locally as shown in Figure1, that will use only the right column for forwarding the subsequent IP packets, see Figure 3. Then the source workstation changes the value of the control field to '11', so that all routers and the destination workstation know that it is the data transmission stage. Accordingly, it inserts the control field with the value '11' and the right side column with the values '4','6','8' into the Ethernet frame. Then, it sends the IP packet to router1, where the same process is performed toward the forward direction till all data is transferred. Note that ROACM implemented at the IP/network/transmission level can contain protocol-specific addresses rather than MAC/hardware addresses. This means that the next hop could be an interface or a logical address.

3. Background of TCP/IP

Protocols in networks are mostly organized in layers; among which every layer is accountable for various sides of the communications. TCP/IP that is the protocol suite combines various protocols at different layers. It is 4-layer system. Link layer known as network interface layer or Network Access and encompass the driver for device in operating system and subsequent card of network interface in computer. The Internet layer or network layer takes care of transmission of packets over networks. The transport layer manages data flow between two hosts. The protocol suite of TCP/IP entails user datagram protocol and transmission control protocol. Application layer takes care of specific application details. In the application layer, the main protocol is the Hypertext Transfer Protocol (HTTP), and there are other application protocols such as file transfer protocol, protocol of simple network management and protocol of simple mail transfer[17].

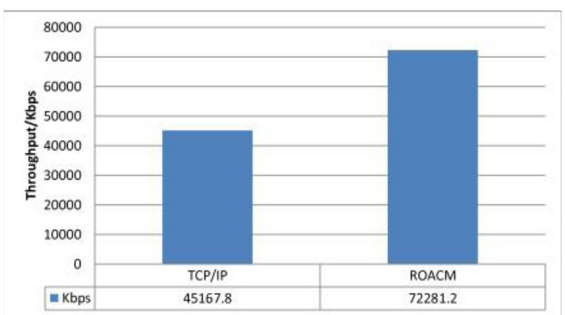

Figure 7. The throughput for the run time 10 seconds.

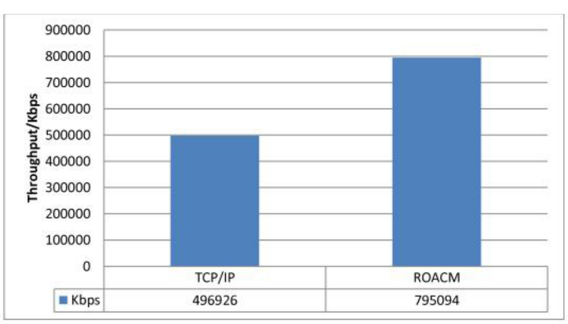

Figure 8. The throughput for the run time 100 seconds.

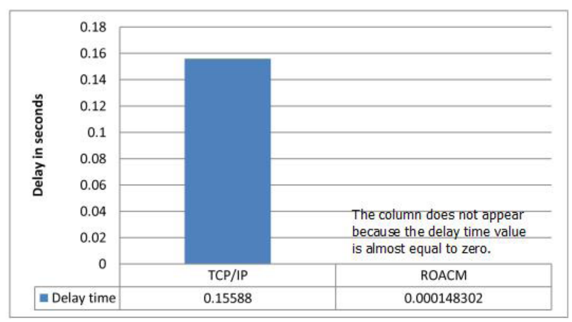

Figure 9. The delay for the run time 10 seconds.

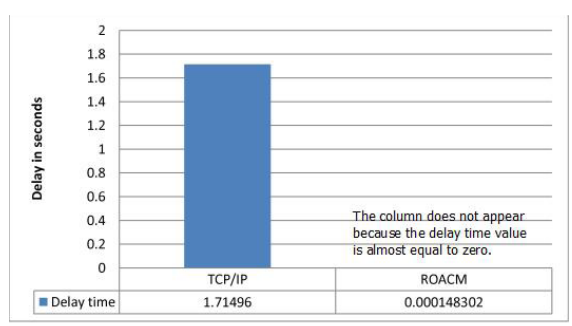

Figure 10. The delay for the run time 100 seconds.

3.1 How a Packet Moves from Source to Destination Work Station

At the source workstation, the data moves down from the application layer to the transport layer where the data is segmented and the transport layer's header is added. Then the segments move down to the Network Layer (NL) so, the header is added including the IP addresses. The packet moves down to the Data Link Layer (DLL), where the header and the trailer are added. Then the packet sent to the router through the physical connection. The router strips off the DLL header and moves the packet up to the NL. The router searches the destination network address in the routing table, and then it forwards the packet to the next router (hop). This process is continued till the packet reaches the destination. The search in the routing table at each hop cause delay depends on the traffic in that router. The RFC 793 and RFC 791 describe the functions performed by TCP and IP respectively.

4. Performance Analysis

This research is based on the comparison between the TCP/IP and ROACM protocols using the network simulation tool (NS3) to study different aspects of performance analysis. Currently, we believe that the important aspects of performance analysis are the average delay, and throughput. To conduct this research, we created two network simulation scenarios one is for the TCP/IP while the other is for the ROACM. Each scenario consists of a source and a destination workstation, and between them are twenty routers. The packet size is 1000 bytes. Each scenario has been run for different period of times (10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 seconds.) We could not control the number of packets generated in each scenario for each period of time. For example, for the period of time, 10 seconds, the number of generated packets in the TCP/IP scenario are 5624 packets while in the ROACM scenario the number of generated packets for the same period of time are 9000. Although more packets were generated in the ROACM scenario, the performance of the ROACM is better than TCP/IP protocol, see Table 1. The table shows all results but we will explain some scenarios in some details. Regarding the throughput, we plotted the throughput at run time 10 and 100, see Figures 7 and 8 respectively. Both figures show that the ROACM protocol outperforms the TCP/IP protocol.

Table 1. The comparison between TCP/IP and ROACM protocols

TCP/IP	ROACM
The run time for 10 second: Tx Packets: 5624 Rx Packets: 5624 Average Delay: 15588 Throughput: 45167.8Kbps	The run time for 10 second: Tx Packets: 9000 Rx Packets: 9000 Average Delay: 0.000148302 Throughput: 72281.2Kbps
The run time for 20 second: Tx Packets: 11874 Rx Packets: 11874 Average Delay: 0.329112 Throughput: 95363.1Kbps	The run time for 20 second: Tx Packets: 19000 Rx Packets: 19000 Average Delay: 0.000148302 Throughput: 152594Kbps
The run time for 30 second: Tx Packets: 18124 Rx Packets: 18124 Average Delay: 0.502343 Throughput: 145558Kbps	The run time for 30 second: Tx Packets: 29000 Rx Packets: 29000 Average Delay: 0.000148302 Throughput: 232906Kbps
The run time for 40 second: Tx Packets: 24374 Rx Packets: 24374 Average Delay: 0.675574 Throughput: 195754Kbps	The run time for 40 second: Tx Packets: 39000 Rx Packets: 39000 Average Delay: 0.000148302 Throughput: 313219Kbps
The run time for 50 second: Tx Packets: 30624 Rx Packets: 30624 Average Delay: 0.848805 Throughput: 245949Kbps	The run time for 50 second: Tx Packets: 49000 Rx Packets: 49000 Average Delay: 0.000148302 Throughput: 393531Kbps
The run time for 60 second: Tx Packets: 36874 Rx Packets: 36874 Average Delay: 1.02204 Throughput: 296144Kbps	The run time for 60 second: Tx Packets: 59000 Rx Packets: 59000 Average Delay: 0.000148302 Throughput: 473844Kbps
The run time for 70 second: Tx Packets: 43124 Rx Packets: 43124 Average Delay: 1.19527 Throughput: 346340Kbps	The run time for 70 second: Tx Packets: 59000 Rx Packets: 59000 Average Delay: 0.000148302 Throughput: 554156Kbps
The run time for 80 second: Tx Packets: 49374 Rx Packets: 49374 Average Delay: 1.3685 Throughput: 396535Kbps	The run time for 80 second: Tx Packets: 79000 Rx Packets: 79000 Average Delay: 0.000148302 Throughput: 473844Kbps
The run time for 90 second: Tx Packets: 55624 Rx Packets: 55624 Average Delay: 1.54173 Throughput: 446730Kbps	The run time for 90 second: Tx Packets: 89000 Rx Packets: 89000 Average Delay: 0.000148302 Throughput: 714781Kbps
The run time for 100 second: Tx Packets: 61874 Rx Packets: 61874 Average Delay: 1.71496 Throughput: 496926Kbps	The run time for 100 second: Tx Packets: 99000 Rx Packets: 99000 Average Delay: 0.000148302 Throughput: 473844Kbps

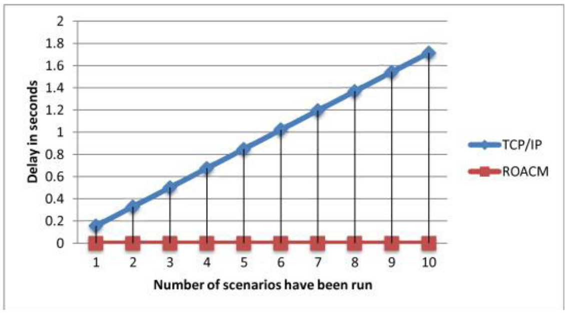

Figure 11. The delay for all scenarios for both protocols.

Regarding the average delay time, we consider the delay in the network layer and neglect the delay in the below layers. Figures 9 and 10 are the plotted average delay time for the run time 10 and 100 seconds, respectively. The delay for the ROACM protocol is almost equal to zero. Also, the delay for 10 seconds scenario is equal to the delay for 100 seconds although the number of pacts sent in the 10 seconds scenario are 9,000 packets while in the 100 seconds scenario are 99,000 packets. This is because in both sessions only the packets in the set up call reach the network layer while the other packets will be cross-connected. Thus, whatever number of packets sent in a session, only the set up delay will be incurred.

Moreover, in Figure 11, we plotted all the values of delay for the tenth scenarios for both protocols. As we mentioned when we discussed Figures 9 and 10 that regardless of the packets transmitted in a session, the delay would be the same in the ROACM protocol but it increased exponentially in the TCP/IP.

The last result shown in Figure 11 is very encouraging to implement the ROACM protocol in today routers and switches since it has the capability to make the routers and switches to cope with the huge traffic, which is still increasing because of the cloud computing, the Internet of the Things (IoT) technologies, and the IPv6 addresses. All these mentioned technologies need a high speed network device.

5. Conclusion

In this researched, we introduce the ROACM protocol and explain the protocol in details while we give a brief introduction about TCP/IP since there are lots of literature explain it. The main intention of this research is to compare the TCP/IP and ROACM protocols. The routing protocol used to build the tables in routers is OSPF, but the comparison is between the TCP/IP and ROACM. The variables selected for studying the efficiency of TCP/IP and developed protocol are average delay and throughput. We built the simulation for ten scenarios for each protocol. From the findings of the simulation outcomes, it was found out that the developed protocol namely ROACM protocol is exponentially more efficient than the TCP/IP protocol. We believe that the ROACM is the solution for the current and next generations for all types of routing/switching devices.

6. Acknowledgement

The author would like to thank Mr. Vimal Ashvini for his help and support to create the scenarios on the network simulation tools NS3.

7. References

1. Ishihara T, Kusuda T, Nagami K, Nakagawa K, Kikuchi Y, Esaki H. A consideration of IX architecture using mpls based on router performance and QoS requirements. IEICE Transaction Communication. 2003 Feb; E86-B(2):498–505.
2. Xiao X, Ni L, Vuppala V. An overview of IP switching and tag switching. ICPADS'97, Seoul: Korea; 1997 Dec 11–13.
3. Rekhther Y, Davie B, Katz D, Rosen E, Swallow G. Cisco Systems' Tag Switching Architecture Overview [Internet]. 1997 Feb. Available from: https://www.ietf.org/rfc/rfc2105.txt.
4. Katsube Y, Nagami K, Esaki H. Toshiba's router architecture extensions for ATM: Overview. IETF RFC 2098. 1997 Apr.
5. Baker F. Requirements for IP version 4 routers. IETF RFC 1812. 1995 Jun.
6. Cisco White Paper. Understanding MPLS-TP and Its benefits. Cisco Public Information.
7. Diab A, Boringer R. Optimized I-MPLS: A fast and transparent micro-mobility-enabled MPLS framework. IEEE; 2006.
8. Xie W, Huang S, Gu W. An improved ring protection method in MPLS-TP networks. IEEE; 2010.
9. Applegate D. Load optimal MPLS routing with N + M labels. IEEE; 2003.
10. Qiu Y. A research of MPLS-based network fault recovery. 2010 Third International Conference on Intelligent Networks and Intelligent Systems; 2010.
11. Newman P, Lyon T, Minshall G. Flow labelled IP: A connectionless approach to ATM. Proceedings IEEE Infocom'96, San Francisco: CA; 1996 Mar. p. 1251–60.

12. Rekhter Y, Davie B, Katz D, Rosen E, Swallow G. Cisco systems' tag switching architecture overview. IETF RFC 2105. 1997 Feb.
13. Newman P, Minshall G, Lyon T, Huston L. IP switching and gigabit routers. IEEE Communications Magazine. 1997 Jan; 35(1):64–9.
14. Kaur G, Kumar D. MPLS technology on IP backbone network. International Journal of Computer Applications. 2010; 5(1).
15. Dumka A, Mandoria H. Dynamic MPLS with feedback. International Journal of Computer Science, Engineering and Applications; 2012.
16. Chan C, Wang P-C, Hu S-C, Lee C-L, Chen R-C. High performance IP forwarding with efficient routing-table update. Computer Communications. 2003 Sep1; 26(14):1641–92.
17. Stevens R. TCP/IP illustrated [Internet]. [Cited 2016 Mar 22]. Available from: http://www.cs.newpaltz.edu/~pletcha/NET_PY/the-protocols-tcp-ip-illustrated-volume-1.9780201633467.24290.pdf.
18. Bahattab A. Route once and cross-connct many; 2010 Feb 16.