

Creating and Managing Dynamic MPLS Tunnel by Using SDN Notion

Hasanein Hasan

Electronic and Computer Engineering
Brunel University London
London, United Kingdom
eepghah@brunel.ac.uk

John Cosmas

Electronic and Computer Engineering
Brunel University London
London, United Kingdom
John.cosmas@brunel.ac.uk

Zaharias Zaharis

Electrical and Computer Engineering
Aristotle University of Thessaloniki
Thessaloniki, Greece
zaharis@auth.gr

Pavlos Lazaridis

Engineering and Technology
University of Huddersfield
Huddersfield, United Kingdom
p.lazaridis@hud.ac.uk

Sinan Khwandah

Maritime, Science and Engineering
Southampton Solent University
Southampton, United Kingdom
sinan.khwandah@solent.ac.uk

Abstract—The aim of this research is using the notion of Software Defined Network (SDN) in creating and managing a Dynamic Multi-Protocol Label Switching (MPLS) tunnel in Open Shortest Path First (OSPF) network. The created tunnel has higher flexibility in terms of allocated path hops and reserved bandwidth. The path of the tunnel can be changed dynamically according to the present requirements of the network as well as its reserved bandwidth which can be changed according to the applied flow data rate. This provides faster path restoration and allows better utilization to the network resources. Moreover, it improves traffic engineering and achieves better Quality of Service (QoS) which results in more reliable network.

Keywords —Real Time Controller; MPLS; OSPF; RSVP; SDN; Traffic Engineering; FDA

I. INTRODUCTION

The Gigantic growth of the internet and the intranet networks creates a series challenge to service providers and equipment suppliers in terms of enormous escalation in traffic [1]. This growth escorts a growth in routing table size.

According to forwarding mechanisms in conventional network layer, when a packet traverses a network, each router extracts all the information relevant to the packet forwarding from the network layer header. This extracted information is used in the routing table lookup operation to detect the next hop for the packet. Each router along the entire path repeats this complicated table lookup operation [2].

MPLS combines the intelligence of routing with the performance of switching. During the entrance of packets to the MPLS domain, labels are attached on the packets, and the label (instead of the IP header) determines the next hop. Labels are taken off at the egress of the MPLS domain [1].

MPLS benefits can be summarized in applications of: Virtual Private Networking (VPN), Traffic Engineering (TE), Quality of Service (QoS) and Any Transport over MPLS (AToM). MPLS reduces the forwarding overhead on the core routers [3].

The MPLS path is known as Label Switched Path (LSP), LSP starts at head-end router and ends at tail-end router [1]. MPLS-TE creates and maintains LSPs by using Resource Reservation Protocol (RSVP) [2]. The Constrained Shortest Path process starts at the head-end router to create LSP by using RSVP-TE messages [1]. The establishment of the LSP tunnel uses two types of RSVP messages: The RSVP-TE PATH and RSVP-TE RESV Messages. The head-end router sends the RSVP-TE PATH Messages to the tail-end router which replies by sending the RSVP-TE RESV Messages which take the same opposite path to the sender.

The routers across the path consider the bandwidth factor in creating the tunnel [4]. If the available bandwidth of any link across the path is not enough to fulfill the requirements of the tunnel, RSVP-TE PATH Message will stop at that point (without carrying on to tail-end router) and a PATH ERROR message will be sent to head-end router without tunnel creation. The router that supports MPLS is called Label Switching Router (LSR) [1].

A. Problems and Challenges

LSRs use routing protocols extensions to create and maintain a TE Link State Data Base (TE-LSDB) [1]. Constraint Based Routing (CBR) algorithm is used to find the best path for an LSP tunnel which also called Constrained Shortest Path First (CSPF) algorithm. CSPF is applied on the head-end router [5]. CSPF uses either the IGP metric or link metric to find the shortest path [6]. Through CSPF, the head-end router uses Traffic Engineering (TE) topology database in finding the path to destination [7].

Although the path discovered to the destination is the shortest, it does not mean it is the best to be used during the current time. Some of the remote links may be occupied by other IP routed flows. Reserving the bandwidth of those links for a MPLS path affects the distribution of other IP forwarded flows and this may cause a link congestion problem.

The bandwidth size of the MPLS Traffic Engineering (TE)

tunnel is automatically adjusted by Cisco IOS MPLS AutoBandwidth allocator. The re-adjustment of tunnel bandwidth depends on the largest average output rate noticed during a certain interval. This involves monitoring the applied flow data rate every interval (several minutes), recording the highest value and using it as a reference to adjust the tunnel bandwidth during the next interval [1].

The tunnel bandwidth re-adjustment is not instantaneous. If the expectation of tunnel bandwidth of the next interval does not meet the real change of flow applied data rate, either the reserved bandwidth will be more than the required (which may consider a waste in using the bandwidth of the network) or the reserved bandwidth is less than required which may cause link congestion.

Tunnel path restoration is performed by the head-end router. At the event of congestion or topology change, the head-end receives notification by RSVP-TE that the path cannot be maintained. Immediately, it constructs a new TE database after removing the faulty links or area of congestion. The operation from the time the router at the place of congestion (or failure) detects the problem until the head-end router finishes the establishment of the alternative tunnel takes from 2-3 s [1]. However, this considered a long time of outage in contemporary networks. If there are several tunnels share an area of congestion and those tunnels were established by different head-end routers, there should be coordination among them on how to deal with their congested tunnels (which of the tunnel paths should be diverted and which should not).

From above discussion, several problems are identified:

- The link congestion problem.
- The delay in tunnel bandwidth re-adjustment.
- The inaccuracy in tunnel bandwidth re-adjustment.
- The delay in path restoration (inflexibility of the path of the tunnel).
- The necessity of coordination among the head-end routers.

B. Contributions and Main Features

Our contribution is a real time pro-active dynamic monitoring controller with supervision and intervention mechanisms. Thus, the controller can:

1. Predict the location of congestion before or at the brink of its occurrence. Remove or mitigate its effect by diverting the flows causing it via other bypass paths.
2. Create a MPLS tunnel, re-adjust its bandwidth instantaneously and accurately, change its path during very short time or remove the MPLS tunnel (if required to do so).

The structure of our conventional MPLS network has been extended to utilize a methodology similar in many aspects to that of the SDN networks (existence of a controller and exchanging of messages between it and the routers). None of the SDN protocols or devices is used as the aim of this research is to develop the performance of the conventional MPLS technology not the SDN technologies.

II. RELATED WORK

MPLS performance development has been studied in traditional routing networking, especially the Open Shortest Path First (OSPF) routing protocol.

All the proposed algorithms that used to dynamically establish MPLS tunnels are applied at the ingress of the tunnel (the head-end router) [8, 9, 10]. This assumes the existence of signaling protocols like RSVP-TE or CR-LDP responsible for the updating of the information related to topology changes and residual bandwidth.

The Minimum Interference Routing Algorithm (MIRA) proposed in [8] tries to decrease the possibility of selecting a route that “interferes” with future requests of other ingress-egress pairs. It uses the residual bandwidth as the main factor on its calculations by identifying the critical links belonging to the other ingress-egress pairs in the network and assigning higher weights for these critical links. It applies Dijkstra algorithm on the graph to avoid the critical links.

Light MIR (LMIR) algorithm is a development of MIRA [9]. LMIR attempts to optimize resource utilization with low computational complexity. Similar to MIRA, LMIR finds paths with lowest capacities to determine the critical edges, assigns weight to them and then executes Dijkstra algorithm to select the non-critical edges.

A Hop-Constrained Adaptive Shortest-Path Algorithm (HCASP) is an algorithm for routing of MPLS bandwidth-guaranteed tunnels proposed in [10]. HCASP has two goals: the first is to choose MPLS path with limit the length so as not to largely exceed the length of the shortest-hop path between the ingress-egress pair; the second is to give preference to less loaded links during the MPLS tunnel creation time to improve the load balance.

All proposed algorithms are applied at the head-end routers which need for a coordination among them to achieve better bandwidth utilization. Our presented algorithm is applied at the controller which provides central management to flows distribution and better bandwidth utilization. Using the controller to create and manage the MPLS tunnel instead of the head-end routers represents the notion of the SDN (separation of the control plane from the data plane).

In [11], Hao et al. presented an algorithm that selects paths for QoS traffic and best-effort traffic. The framework of traffic engineering consists of: optimal computing, getting the path set from the result of the optimal computing and on-line routing which selects paths for QoS traffic. The on-line routing algorithm selects narrowest shortest paths for QoS traffic and light load path for the best-effort traffic.

In [12], Tu et al. suggested a method of splicing MPLS and OpenFlow tunnels based on SDN paradigm. A global view controller provided with information collector, path translator and command installer is responsible of generating MPLS label forwarding rules and OpenFlow entries inside the routers.

Establishment of MPLS tunnel that traverses multiple OSPF areas was discussed in [13].

III. NETWORK MODEL

In this section, the designed network is presented. The network has been designed by using OMNeT++ simulator through modifying its Inet-OMNeT project.

A. Network Model and Network Messages

Our project examines two topologies of an OSPF routing network. The first consists of 9 routers and 7 hosts as shown in Fig. 1 while the second consists of 13 routers and 13 hosts as shown in Fig. 7. Before delving into the network operation details, the types of messages that travel across the network are as follows:

- Flow status updating messages issued by the head-end routers and directed to the controller. They carry the flow current information like flow source, flow destination, flow data rate change and tunnel establishment request. The flow status messages provide the real time updating to the controller.
- Control messages issued by the controller and directed to specific routers.
- OSPF standard messages.
- RSVP standard and developed messages.
- Data packets issued only by hosts and directed to other hosts. Data messages represent the flows and they are defined as User Datagram Protocol (UDP) packets.

All the above mentioned messages and packets traverse the routers of the network to reach to their destinations.

B. The Router Behavior

The router used represents the standard OSPF/MPLS router of the inet-OMNeT project. It is developed with modifications in code and several added models to be combatable with the new design requirements.

1) Router forwarding mechanisms

The router builds its OSPF routing table by exchanging information with its neighbors and responds to the commands of the controller in terms of IP and MPLS forwarding mechanisms. Routers use OSPF routing table to forward the IP routed flows.

2) RSVP model inside the router

The RSVP model inside the router has been developed to be compatible with the RSVP command packets sent by the controller and to perform extra duties. The development is as following:

- The developed RSVP model inside the router will no more read its information from xml file as the main design of the inet-OMNeT.
- No RSVP-TE PATH messages are used to establish the tunnel.
- RSVP-TE RESV messages are developed to substitute the duty of RSVP-TE PATH messages as well.
- RSVP FinalHop_Resv and RSVP Remov_Tunl messages are new types of messages that established and used in the developed RSVP models.

C. The Controller Behaviour

The controller is a pro-active real time device that has the responsibility of monitoring and managing the traffic across the network. It is capable of dealing with any congestion problem. It is connected to some routers. The routers that are not directly connected to the controller use the OSPF routing tables to contact the controller and vice versa. The controller has three main features:

1) Knowledge

At the beginning of the simulation, the controller receives all the information related to the network like topology, link bandwidths and OSPF routing tables of all routers to all destinations. According to this knowledge, the controller constructs the network topology. Therefore, it can predict the behaviour of the routers (forwarding decision of data packets) when they deal with the applied flows. The controller calculates all possible paths from all sources to all destinations and stores them in a list from the shortest to the longest path regarding the OSPF path gotten from each router.

2) Flows Distribution Algorithm (FDA)

The controller is provided with a heuristic traffic management Flows Distribution Algorithm (FDA). It applies the FDA to solve the congestion problem and to calculate the MPLS paths. The controller deals with the congestion by ordering specific routers to change some of their forwarding decisions. FDA takes in consideration the distribution of both IP routed flows and MPLS routed flows. It uses residual bandwidth and hop count as path calculation factors. FDA will be explained in details in another research paper which to be published later.

3) Ability to affect the network

There are two types of control messages issued by the controller: The IP control messages that deal with network layer by creating temporary flow table inside some routers (this work will be explained in details with the operation of FDA in another research paper as mentioned before) and the RSVP control messages that deal with MPLS model to manipulate MPLS tunnel. The controller creates its RSVP control messages inside its developed RSVP model which contacts and affects the developed RSVP model of the router. The controller deals with MPLS according to one of four actions:

- *Creating a MPLS tunnel*
- *Re-adjusting the reserved bandwidth of the MPLS tunnel*
- *Removing the MPLS tunnel.*
- *Changing the path of the MPLS tunnel.*

The first and third actions are applied according to a request from the sender host while the second and forth actions are applied according to the network requirements. The problems discussed in the introduction section are solved as follows:

- The link congestion problem has been solved by using the central controller that predicts congestion and

selectively manages flows according to FDA.

- The controller is able to re-adjust the reserved bandwidth of the tunnel immediately and exactly according to the applied flow data rate.
- The tunnel can be replaced with another tunnel that passes through another path and serves same source – destination pair if the existing path is not suitable currently. The creation time of the new tunnel path is short.
- There is no need for coordination among the head-end routers as the tunnel establishment and maintenance operations are the responsibility of the controller.

IV. PERFORMANCE

For every new flow (or obvious change in the data rate of an existing flow), the head-end router will normally deal with the flow by sending a flow updating status message that contains the flow information to the controller.

The sender host (the source) of the flow requests the head-end router to create a MPLS tunnel to destination. The head-end router inserts the request of tunnel inside the flow status message.

When the controller receives the flow status message of the head-end router, it will activate its FDA. The FDA models the network, examines the flow mimicry distribution over the network and allocates a specific path to serve the flow. The controller deals with MPLS according to one of four actions:

1) Creating a MPLS tunnel

The path of the tunnel is chosen according to the decision of the FDA inside the controller. The tunnel creation procedure is as follows:

1. Allocate the tunnel attributes represented by tunnel path list, tunnel bandwidth, and flow source and flow destination. Select specific numbers for tunnel ID, tunnel LSP ID and tunnel color. Store all the mentioned information into specific container.
2. Insert tunnel attributes into RSVP FinalHop_Resv message (which is created in the developed RSVP model inside the controller), encapsulate it with IPv4 datagram and send it to the tail-end router of the tunnel.
3. The tail-end router network layer will de-capsulate the arrived IPv4 FinalHop_Resv message then send it to its router developed RSVP model.
4. The router developed RSVP model extracts the tunnel attributes from FinalHop_Resv message and use them to create Explicit Route Object (ERO), Record Route Object (RRO), Path State control Block (PSB) list and Reservation State control Block (RSB) list.
5. Upon receiving RSVP FinalHop_Resv message, the router developed RSVP model creates internal timer messages RsbCommitTimerMsg and RsbRefreshTimerMsg.
6. After receiving RsbRefreshTimerMsg, the router developed RSVP model creates RSVP-TE RESV

Message, inserts tunnel attributes into it and sends it to the pen-ultimate router of the path list.

7. The pen-ultimate router receives RSVP-TE RESV Message, repeats steps 4, 5 and 6 above and sends RSVP-TE RESV Message to the next previous router in the MPLS path list. This operation is repeated at each router through the MPLS tunnel path until reaching the head-end router where no more RSVP-TE RESV Messages issued.
8. At the head-end router, the tunnel establishment operation finishes. Tunnel attributes are also added to SimpleClassifier model to build the Label Information Base (LIB) table. This enables MPLS model of the head-end router to push a label into the flow packets entering the MPLS domain.

e.g., suppose a network of 9 routers with topology as shown in Fig. 1. Host B starts sending flow to host E. host B wants flow B-E passes through MPLS tunnel. Host B starts sending flow B-E data packets to router 2 implying its request inside the flow packets. Router 2 sends flow status message to the controller illustrating flow source, flow destination, flow data rate and the MPLS tunnel establishment request.

The controller applies its FDA upon the flow characteristics. The result is creating the MPLS tunnel that passes through routers 2, 3, 1 and 5. This path represents the OSPF distribution path as there is no congestion over it. The controller sends RSVP FinalHop_Resv message to router 5 (which is the tunnel tail-end router) as shown in Fig. 1.

Router 5 records the attributes of the MPLS tunnel and sends RSVP-TE RESV Message to router 1 (which is the tunnel pen ultimate router). The operation is repeated across the tunnel path until reaching the head-end router (router 2) where the establishment of the MPLS tunnel completes successfully. Flow B-E is forwarded over the established MPLS tunnel path passes through routers 2, 3, 1 and 5.

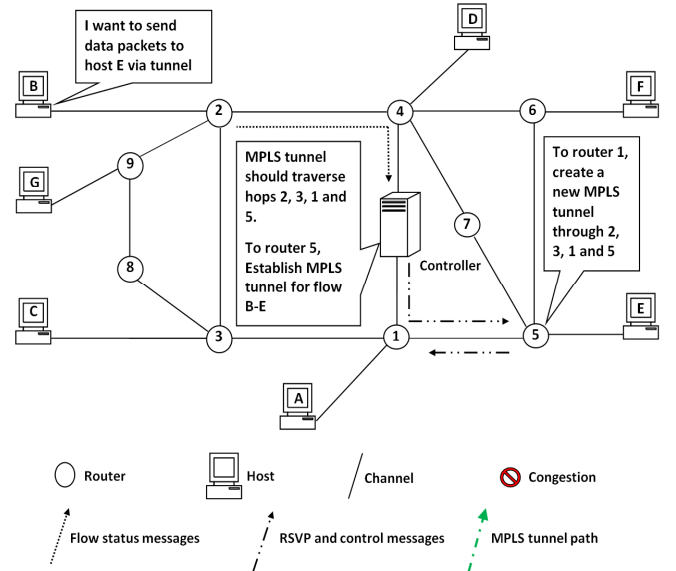


Fig. 1. The process of establishment of MPLS tunnel in a network consists of 9 routers.

2) Re-adjusting the reserved bandwidth of the MPLS tunnel

If the data rate of the applied flow across the tunnel is changed (more or less 5% from its previously measured value), the head-end router informs the controller through flow status message about the new required bandwidth for the tunnel. Upon receiving the flow status message, the controller applies its FDA that internally examines congestion on links.

If there is no congestion, there is no need to change the path of the existing MPLS tunnel. The FDA model repeats the same procedure it used before in creating the tunnel. The new created tunnel will take place the old tunnel which is removed from the path.

The new tunnel is compatible with the old one in path, flow source and flow destination but different from it in other tunnel attributes such as reserved bandwidth, tunnel ID, tunnel LSP ID and tunnel color.

e.g., suppose that in the network depicted in Fig. 1, the data rate of flow B-E increased obviously. Router 2 informs the controller about the flow status through flow status message.

The controller applies its FDA upon the flow characteristics. The result is using the same path of the existing MPLS tunnel that passes through routers 2, 3, 1 and 5 but with different reserved bandwidth. The controller orders establishing a new MPLS tunnel by inserting its attributes in a RSVP FinalHop_Resv message and sending the RSVP FinalHop_Resv message to router 5 (which is the tunnel tail-end router) as shown in Fig. 2.

Router 5 records the attributes of the new MPLS tunnel, removes the old MPLS tunnel and sends RSVP-TE RESV message to router 1 (router 1 is the tunnel pen ultimate router).

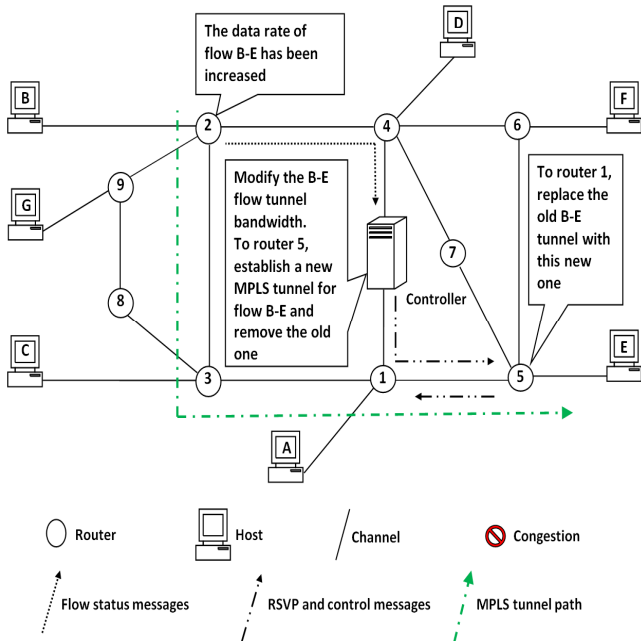


Fig. 2. Re-adjusting the reserved bandwidth of the MPLS tunnel by replacing the old tunnel with a new one of same path.

The operation is repeated across the tunnel path until reaching the head-end router (router 2) where both the establishment of the new MPLS tunnel and the removal of the old MPLS tunnel complete successfully. Flow B-E continues using the path passes through routers 2, 3, 1 and 5 but with higher reserved bandwidth as shown in Fig. 2.

3) Removing the MPLS tunnel

The sender host may request the controller to remove the used MPLS tunnel. The controller takes the following steps:

- The controller creates a Remov_Tunl message, inserts tunnel's ID and path inside it and sends it to the tail-end router.
- The tail-end router receives the Remov_Tunl message, removes the tunnel from the RSVP model, creates another Remov_Tunl message and sends it to the pen ultimate router.
- The operation is repeated until reaching the head-end router where the MPLS tunnel is completely removed.

After tunnel removal, routers forward flow packets according to the routing table of network layer.

e.g., suppose that in the network depicted in Fig. 2, where host B wants to remove the tunnel to host E. Host B implies its request inside the flow packets. Router 2 receives the request then sends a flow status message to the controller. The controller sends Remov_Tunl message to router 5 (the tunnel tail-end router).

Router 5 starts removing the old MPLS tunnel passes over routers 2, 3, 1 and 5 through sending Remov_Tunl message to router 1 and so on until reaching the head-end router (router 3). The illustrated operation is depicted in Fig. 3.

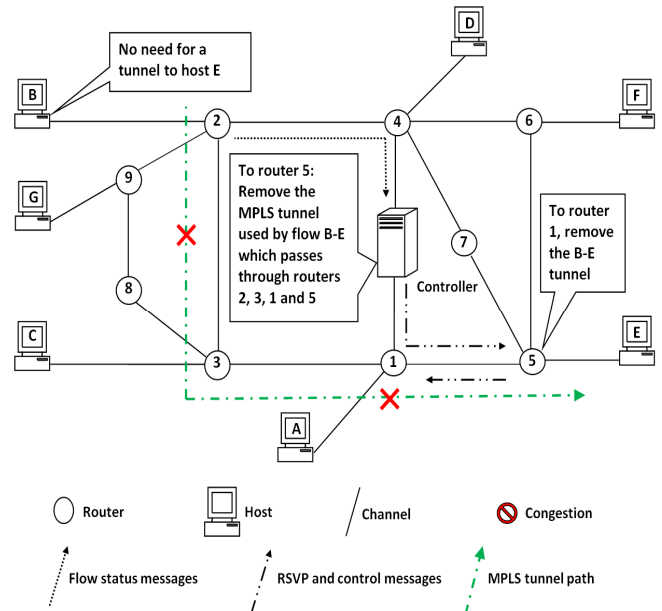


Fig. 3. The process of removing of MPLS tunnel.

4) Changing the path of the MPLS tunnel

If the data rate of the applied flow across the tunnel is increased obviously the head-end router informs the controller through flow status message the new bandwidth required for the tunnel. Upon receiving the flow status message, the controller applies its FDA and predicts congestion starts occurring somewhere on the path. The controller decides to change the path of the tunnel to avoid the congestion. The controller uses both scenarios of creation and removal of MPLS tunnel mentioned in cases 1 and 3 illustrated above. It creates the new tunnel over the new suggested path (where there is no expected congestion). At the same time, the controller removes the existing old tunnel. The flow starts using the new established tunnel.

e.g., suppose that in the network depicted in Fig. 2, the data rate of flow B-E increased significantly. Router 2 informs the controller about the flow status through flow status message as usual.

The controller applies its FDA upon the flow characteristics. The result of the FDA shows that there is congestion over link 1-5 as shown in Fig. 4. Therefore the MPLS tunnel path should be changed to prevent the congestion. The new (detour) MPLS tunnel (proposed by the FDA) passes through routers 2, 4, 6 and 5 instead of the old MPLS tunnel which passes through routers 2, 3, 1 and 5.

The new path represents an alternative path from the OSPF path due to the congestion occurred upon the latter one.

The controller orders to establish the new MPLS tunnel by inserting its attributes in a RSVP FinalHop_Resv message and to remove the old MPLS tunnel by inserting its ID and path in a RSVP Remov_Tunl message. The controller sends Both FinalHop_Resv and Remov_Tunl messages to router 5 (the tail-end router).

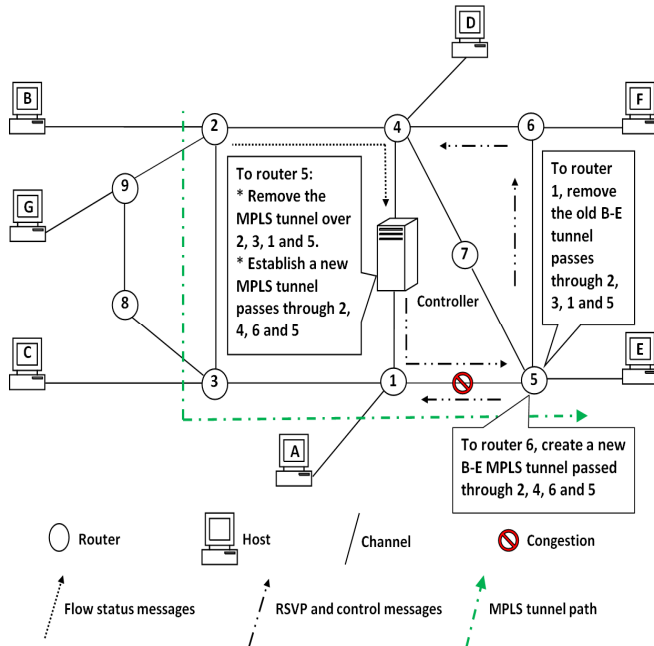


Fig. 4. The process of changing the path of MPLS tunnel.

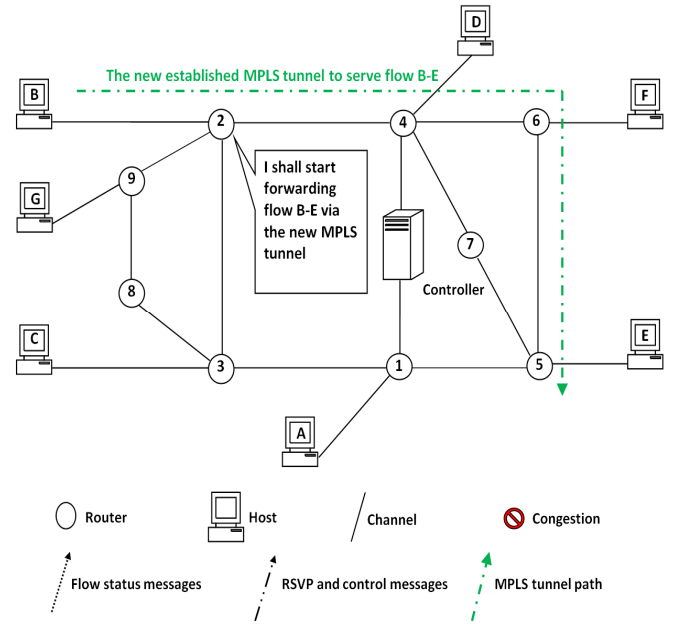


Fig. 5. The B-E flow passes over the new MPLS tunnel path.

Router 5 repeats the scenario of creating new MPLS tunnel over routers 2, 4, 6 and 5 through sending RSVP-TE RESV message to router 6 and so on. Router 5 starts removing the old MPLS tunnel passes over routers 2, 3, 1 and 5 through sending Remov_Tunl message to router 1 and so on. Both the creation and removal operations continue until reaching the head-end router (router 2). The illustrated operation is depicted in Fig. 4. Flow B-E starts using the new MPLS tunnel path as show in Fig. 5.

To test the tunnel creation time over larger networks of more complicated topology, this methodology has been applied on the network shown in Fig. 7 (UK Map) where there are 13 routers and each router is connected to a host that represents a city in the United Kingdom. The hosts apply more variable traffic values sent to different destinations.

During the first 30 minutes of the simulation, the hosts commence applying load to the network. The controller establishes the MPLS tunnels from the head-end routers to the tail-end routers as required.

During the next 15 minutes of the simulation, some hosts increase their applied data rates. The controller applied its FDA to establish new tunnels with higher bandwidth. Some of new tunnels take the same paths of their ancestors and other take different paths to avoid congestion. During the third 15 minutes of the simulation, the hosts which sending high data rates reduce their applied load to moderate values. The controller returns the diverted flows to their original paths.

Finally and during the last few minutes after the first hour, hosts continue applying the same moderate data rate values where there is no need to establish or change the path of any MPLS tunnel. The total simulation time of both topologies (the network of 9 routers and that of 13 routers) lasts for one hour and several minutes.

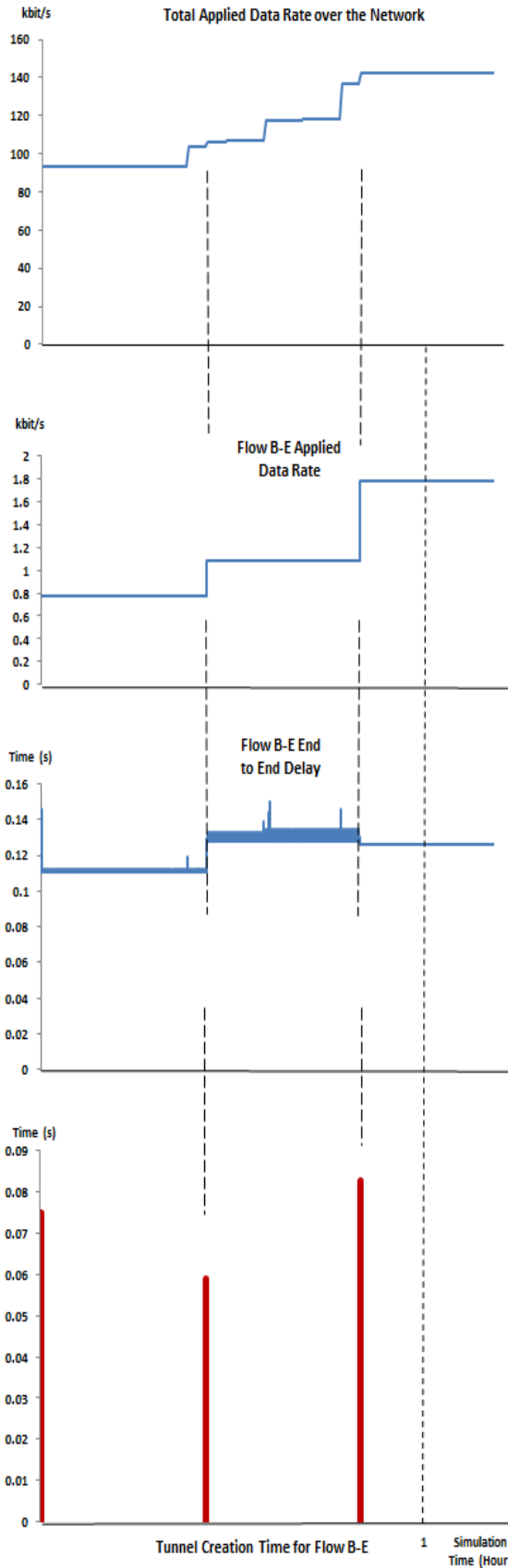


Fig. 6. End to end delay and tunnel creation time for flow B-E along with flow B-E applied data rate and total applied data rate of 9 routers network



Fig. 7. The network of 13 routers (UK map)

V. RESULTS

The results shown in Fig. 6 represent the tunnel creation time duration for flow B-E of the 9 routers network during three different periods of simulation time that meets three different applied load values. The first tunnel creation time value was recorded when all flows started. At that time, the controller was busy with many mathematical operations in calculating the paths related to the flows as it was receiving many status messages that included many requests to create tunnels. The second creation time value of tunnel was recorded during moderate applied load over the network while the third value was recorded during high applied load. The tunnel creation time of flow B-E and the network applied load status are also shown in Table I.

TABLE I
TUNNEL CREATION TIME

Time duration of tunnel creation	Applied load status	Controller action
74.9 ms	Start	Tunnel first creation (via main path)
58.9 ms	Moderate	Tunnel bandwidth updating (via main path)
83 ms	High	Creating a tunnel of an alternative path (restoration by a detour path)

It can be noticed that tunnel creation time for all cases is small value compared to the requirements of modern communications. End to end delay (Latency) for flow B-E is also an acceptable value.

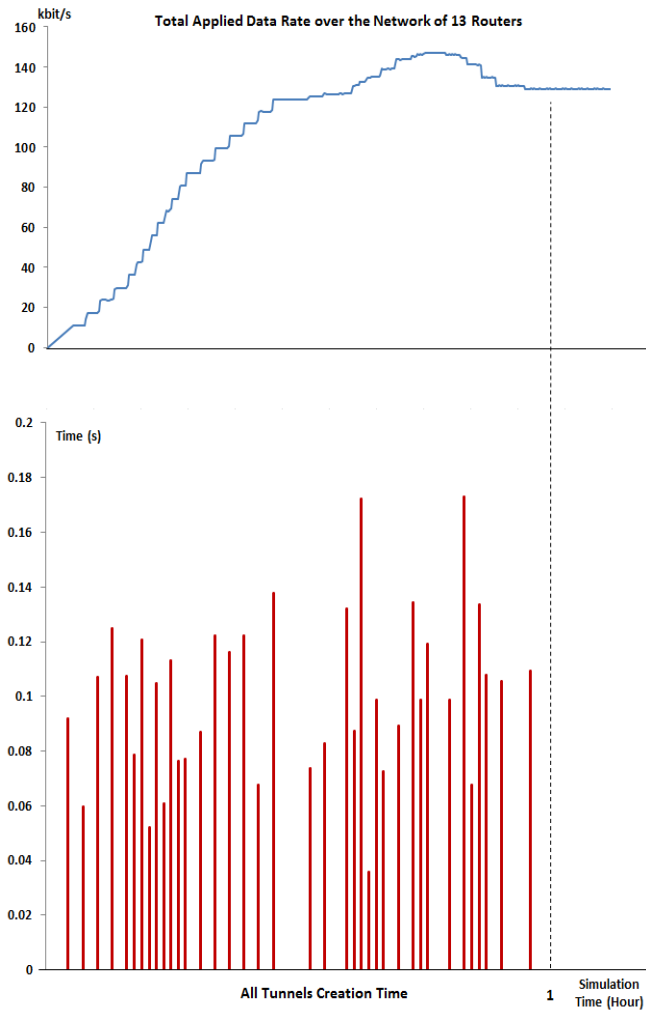


Fig. 8. All tunnels creation time along with total applied data rate of UK map network of 13 routers

The results shown in Fig. 8 represent the time durations recorded to create or restore (change of path) of all the tunnels in the simulation of network of 13 routers shown in Fig. 7. The longest value of time that was recorded to create the longest replacement (detour) MPLS tunnel path (Newcastle – Edinburgh - Glasgow - Belfast – Liverpool) under high applied load is 173 ms.

VI. CONCLUSION

The dynamic establishment and management of the MPLS tunnel by the controller have provided better utilization to the available bandwidth. By fast updating the tunnel requirements, flow that traverses the tunnel suffers less from queueing delay and gains less packet drop. As the controller knows the flows distribution and the residual bandwidth over all network links, tunnel creation time and tunnel restoration time are reduced as tunnel creation is easier and there is no possibility of PATH ERROR messages to exist.

Tunnel creation time is affected by several factors like hops number, applied load value, applied load pattern, the degree of the complexity of the topology and the distance away from the

controller which also has a little effect. The network takes longer time to create or change the path of the MPLS tunnel if the new path passes via more number of hops or there is high applied load over the network. However, it is still much less than the value declared in [1] which ranges between 2-3 s. If the controller tries to establish several tunnels at the same time, this may cause small extra delay as well.

The result is a flexible MPLS tunnel in the terms of path and reserved bandwidth. The path of the established tunnel that serves same source-destination may pass completely or partially over the main OSPF routing table path or pass through very different path as the head-end and tail-end routers are the common hops among all available paths. The tunnel created by the controller has a global identification number.

REFERENCES

- [1] Cisco Systems. Advanced Topics in MPLS-TE Deployment. [Online]. Available: http://www.cisco.com/c/en/us/products/collateral/ios-nx-oss-ware/multiprotocol-label-switching-traffic-engineering/whitepaper_c11-551235.html
- [2] Cisco Systems. Multiprotocol Label Switching Overview. [Online]. Available: http://www.cisco.com/c/en/us/td/docs/ios/12_2/switch/configuration/guide/fswtch_c/xcftagov.html
- [3] Cisco Systems. MPLS FAQ For Beginners, Document ID: 4649. [Online]. Available: <http://www.cisco.com/c/en/us/support/docs/multiprotocol-label-switching-mpls/mpls/4649-mpls-faq-4649.html>
- [4] L. D. Gheini, "MPLS Fundamentals," 6th ed., Indianapolis: Cisco Press, 2007, pp. 279.
- [5] Cisco Systems. How MPLS Traffic Engineering works. Cisco Support Community. [Online]. Available: <https://supportforums.cisco.com/document/10431/how-mpls-traffic-engineering-works>
- [6] Cisco Systems. MPLS Traffic Engineering, Constraint-Based Routing and Operation in MPLS TE. Cisco Press. [Online]. Available: <http://www.ciscopress.com/articles/article.asp?p=426640&seqNum=3>
- [7] Cisco Systems. MPLS Traffic Engineering: Inter-AS TE. [Online]. Available: http://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/guintast.html
- [8] Kar, K.; Kodialam, M.; Lakshman, T.V., "Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications," in *Selected Areas in Communications, IEEE Journal on*, vol.18, no.12, pp.2566-2579, Dec. 2000.
- [9] Figueiredo, G.B.; da Fonseca, N.L.S.; Monteiro, J.A.S., "A minimum interference routing algorithm," in *Communications, 2004 IEEE International Conference on*, vol.4, no., pp.1942-1947 Vol.4, 20-24 June 2004.
- [10] Elsayed, K.M.F., "HCASP: A hop-constrained adaptive shortest-path algorithm for routing bandwidth-guaranteed tunnels in MPLS networks," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, vol.2, no., pp.846-851 Vol.2, 28 June-1 July 2004.
- [11] Kun Hao; Zhigang Jin, "An On-Line Routing Algorithm Based on the Off-Line Optimal Computing in MPLS," in *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, vol., no., pp.1-5, 24-26 Sept. 2009.
- [12] Xiaogang Tu; Xin Li; Jiangang Zhou; Shanzhi Chen, "Splicing MPLS and OpenFlow Tunnels Based on SDN Paradigm," in *Cloud Engineering (IC2E), 2014 IEEE International Conference on*, vol., no., pp.489-493, 11-14 March 2014.
- [13] Chongwen Wang; Jing Hu, "Improvement of Running Tunnel Based on OSPF TE," in *Web Information Systems and Mining (WISM), 2010 International Conference on*, vol.2, no., pp.3-7, 23-24 Oct. 2010