A Joint Approach to Multipath Routing and Rate Adaptation for Congestion Control in OpenFlow Software Defined Network

Sofia Naning Hertiana
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Jl. Ganesa No.10 Bandung, Indonesia
School of Electrical Engineering
Telkom University
Jl. Telekomunikasi No. 1 Bandung, Indonesia

Hendrawan, Adit Kurniawan
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Jl. Ganesa No.10 Bandung, Indonesia
adit@stei.itb.ac.id,hend@stei.itb.ac.id

***Abstract*—Software Defined Network is a new paradigm in data communication networks. In contrast to the traditional internet network, SDN separates the control function and the function forwarding. SDN controller operates on the principle of a centralized logical controller. This centralized control causes the controller to have a global network knowledge so that SDN can be more effective in managing the network. In this paper, we apply a congestion control mechanism by utilizing the global knowledge possessed by the controller. Congestion control mechanisms implemented by combining multipath routing mechanisms and rate adaptation in an integrated manner. The controller uses the information from the switches that are used to determine the route along the source and destination, as well as to determine the appropriate rate for each flow. The experimental results show that the combination of routing and rate adaptation can effectively be used to control congestion, resulting in a better performance than single path routing and multipath routing without rate adaptation.**

Keywords*—*software defined network; multipath routing; rate adaptation

I. INTRODUCTION

SDN is a new paradigm in network technology. Unlike traditional internet network that combines control plane and data plane in the same node, SDN separates the control plane and data plane. This separation makes the network policy can be done by software in the control plane. Control plane consists of one or more controllers are connected to the switches in the data plane. Separation of these functions makes the data plane on SDN is only a collection of dumb devices [1]. On SDN, the communication between the control plane and the data plane using a protocol. The protocol was first used is OpenFlow [2] and this protocol is open.While not everyone believes that OpenFlow is already mature enough to support future internet and resolve the real problems the network, but until now OpenFlow can be used to solve the challenges of the future internet experimentally. OpenFlow approach provides easy access to the flow table, providing real-time control via network switches and allows administrators to monitor and control the route packets flowing through the network [3].

National Science Foundation predicts that the Internet will have nearly 5 billion users by 2020. Other data mentions that the user Internet of thing (IOT) reached 7,065 billion. A very large number of customers have become one of the serious problems in the future internet [4]. The future internet network requires the availability of sufficient network resources and network management is good, otherwise it will cause network congestion. The adverse effects of network congestion are the decline in performance, such as high delay, low throughput, and packet loss [5]. This can reduce customer satisfaction and will harm the network and Internet service providers. Therefore, congestion must be controlled to maintain network performance remains excellent.

Network congestion problems begin to be solved since 1988 by Van Jacobson with the proposed TCP [6]. TCP uses the settings window (sending window) to adjust the sending rate. Traditional window-based congestion control does not work well when dealing with bursty traffic, resulting in additional queuing delay and packet loss. Besides that TCP has some limitations and drawbacks that are prone to instability, it is not fair to the connection with the high round-trip delay and low resource utilization. Already many of the proposed scheme for TCP improvements, among others set forth in [7] [8] [9] [10] [11]. Proposed improvements TCP has also been done on SDN, among others OpenTCP [12], CCC (Cooperating Congestion Control) [13] and in [14]. The proposed improvement is shown to improve network performance, but congestion control that is not supported by either routing mechanism making it ineffective.

On the other hand, there are some ideas to solve the network congestion by controlling routing. Routing mechanisms on the Internet network is largely based on a single path routing. However, routing with a single trajectory contributing to congestion on the network due to the possibility of some of the same links used as the shortest path from a different node pairs that causes convergence to link the same package [15], [16]. Improvements single routing path is the multiple path routing techniques (multipath routing) that are used as a method to reduce network congestion. The basic idea

978-1-4673-8434-6/15/$31.00 ©2015 IEEE

of multiple routing paths is to divide the package with the same origin and destination for some of the alternative pathway. The use of multipath routing proven to improve network resource utilization [17].

The use of the routing mechanism and a separate delivery rate setting is not effective enough to prevent network congestion. Combining multipath routing by setting the rate of delivery can be effective for congestion control and can improve performance better than if applied separately. Routing determines the path to be used while congestion control determines what rate to be sent through any predetermined path.

On SDN, routing and sending rate settings can be fully controlled by the controller. The controller has a global network of knowledge about the condition periodically based on information reported by the switch to decide the action to be taken. With these advantages, it is possible to develop a scheme of a combination of routing and rate adaptation to control congestion on OpenFlow [3]. By using the advantages offered by SDN OpenFlow, we applied a combination of multipath routing scheme and setting the pace of delivery with the aim to control network congestion and improve network performance. To achieve these objectives, we have proposed a design that contains components that function as the search for the best path and set the rate to be allocated on each path. In fact, OpenFlow 1.3 [18] already has support for (ECMP), but Al-Fares et al. In his paper [19] found the collision on ECMP that occur in the long term can cause a lot of lost bandwidth.

The rest of the paper is structured as follows. Section I. Introduces, section II Related work, section III Design, in section IV Evaluation and the last in Section V Conclusions and future work.

II. RELATED WORK

Some literature have discussed multipath path routing of different purposes, e.g. [20], [21], [22], [23] explains how to find a loop-free multipath routing. Two others, such as [24], [25], discussing the multipath routing optimization problem, finding the optimal way to pass traffic with high throughput and low congestion. Routing path as QoS multipath routing discussed in [16], [26], [27], [28], [29] wherein multipath routing used to meet the QoS guarantees. Multipath routing as load balancing methods are discussed in [30], [31], [32]. Likewise, within the scope of SDN, multipath routing has been discussed in [33], [34].

Multipath routing schemes that have been proposed as a whole can improve network performance. Multipath routing produces a better performance than the single path routing scheme. However, the schemes are still not effective. The effectiveness of a multipath routing scheme can be improved by combining multipath routing and rate adaptation of the predetermined path of the routing algorithm. Merging multipath routing by setting the sending rate for optimization purposes proposed by Fern Y. Hunt [35], Roberto Commineti [36], and Jiayue [37]. Most of the proposed method is done for traditional internet platform. Nodes in the network exchange information with each other and not doing congestion centrally. This system has a significant delay constraint resulting feedback to the adaptation rate to be expired. At the data center, the design of multipath algorithms has been proposed by Al Fares in [19], Heller [38] and Benson [39]. On the other hand, in [40] [41], the problem of multi-path QoS can be formulated as a Multi-Commodity Flow (MCF). Combining routing and rate adaptation on SDN proposed in [14]. They combine routing with explicit feedback congestion control. The proposed scheme generates flow average completion time (AFCT) is small. However, this scheme uses a single path routing and does not provide an alternative path in the event of overload.

III. DESIGN

Before discussion of the combination of multipath routing mechanisms and rate adaptation, it is helpful to consider the mechanism of how the flow to be processed in SDN OpenFlow. When the first packet of a new flow arrives at the switch, the switch will see the flow table to determine how the packet should be processed. If the package is not found in the existing rules, the switch, then encapsulates the packet to the Packet-inclusive message and sends the message to the controller. After receiving a message packet-In, OpenFlow controller determines how the flow should be handled. Handling flow based on policy settings is set and the network conditions globally. If this flow is permitted, the controller calculates the flow path and install a new entry corresponding flow switch along the path. The controller may flow modification command to switch on each switch along the route or may also send a message Packet-Out that explicitly tells the switch where to forward the packet first.

The scenario of a combination of multipath routing and rate adaptation in this paper begins when there is a request from the source node to the destination node. OpenFlow switches will be reported to the controller about the state of the link. The controller will collect and calculates the network conditions. The controller will determine all possible paths that will be used to pass flow from source to destination. The controller also calculates the rate that can be allocated to each flow is passed. The combination of multipath routing mechanism and the adaptation rate on OpenFlow SDN that we propose as shown in figure 1.

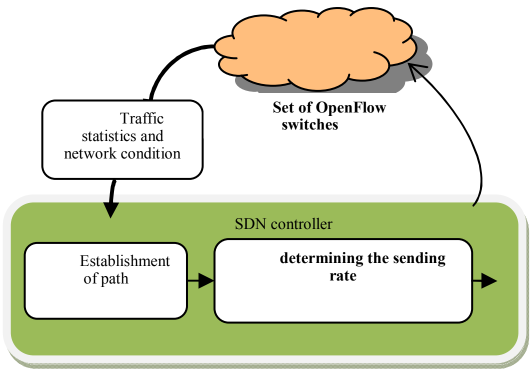

Figure 1: The mechanism of combination of multipath routing and rate adaptation

In general, the steps are as follows:

- Step 1: controller collects information on network conditions
- Step 2: controller finds the best path and several alternative paths from source to destination
- Step 3: the controller calculates rate that can be allocated in accordance with the utilization of a path
- Step 4: the controller ordered the switch to pass the traffic through the best path
- Step 5: if congestion marked path utilization exceeds the threshold, then the flow is distributed to an alternative route until a certain threshold

A. Network Model

Network modeled as a simple directed graph G (N, L). Where N is the node and L is the link. (i, j) is the pair where i is the outgoing node and j is the incoming node. If *s* is the source node and *d* is the destination node, then the entire route from source to destination is the entire path from the source to the destination or P_{sd} . Where a path from source to destination is expressed by equation (1).

$$P = \{s, i_1 . i_2 \ldots \ldots., d\} \quad (1)$$

$$P \in P_{sd}$$

B. Selection of the shortest path

In general, any routing protocol used in packet networks such as OSPF (Open Shortest Path First) is used as a link metric weight. Based on the weight of the link, routing protocols find the path end to end with the most minimal total weight for each pair of source and destination. This mechanism is called the path selection by choosing the shortest route "shortest path routing". In this paper, we use the DFS (Depth-first search) algorithm to search routes. All possible routes (paths) are stored in the flow table T and compared with each other. This comparison is to find a path with the maximum residual bandwidth. Path discovered regarded as the best path and then some other path as an alternative route.

$$p = max\ \{f_{resBW}(p) | p \in P_{sd}\} \quad (2)$$

The capacity of a path (C_p) is the minimum capacity of a link that is a member of a path:

$$C_p = min\ \ \{C(l) | l \in p) \quad (3)$$

So that the residual bandwidth of a path ($Res\ bw$) is the minimum capacity is reduced by a rate that is already in use.

$$Res\ bw = C_p - s(t) \quad (4)$$

C. Distribution Traffic and Threshold Path

Traffic distribution is done when the bandwidth utility the main path through which the flow has reached a certain threshold. We use a bandwidth utility as a parameter to divert flow from the best route as well as the main route to the alternative route. Traffic was first passed to the best path and any particular period controller update path status. If the utility bandwidth on the main path reaches 0.8, the controller will divert traffic from the main road to the alternative path.

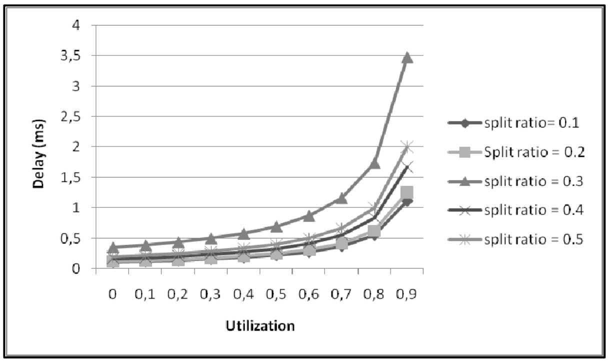

Figure 2: Relationships delay and bandwidth utilization in OpenFlow switch using the model Jackson

In this case, we use the threshold is 0.8 as the threshold for traffic diversion. Reason why the use value of 0.8, we can see in figure 2 that the delay has not increased significantly when the utility to the value of 0.4. Delay increases linearly when the utility reaches 0.4-0.8 and increase dramatically when the utility reached above 0.8. Delay increment can be used as a parameter that the path being traversed a flow is experiencing congestion. The relationship of the delay and bandwidth utilization in figure 2 is obtained from the equation 5.

$$T_s = \frac{\rho}{\lambda(1-\rho)} \quad (5)$$

$$\rho = \frac{\frac{\lambda}{1-q}}{\mu} \quad (6)$$

Where T is a delay, λ is the average traffic coming into the switch, ρ is bandwidth utilization and q is the split ratio (incoming packet to the controller as a new flow).

D. Rate adaptation

Rate adaptation is used to adjust the data transmission rate of the link load conditions during the routing process. Most of rate link adaptation is not based on this path as a whole. In this paper, rate adjustments made to find the maximum rate path for any paths that have been found. Rate adjustment is based on the condition of the path that has been chosen.

The rate is allocated according to the residual bandwidth is used to determine a path. The rate is updated every interval d, then for each flow that will be serviced using the rest of the path capacity.

$$R(t) = \frac{\alpha\,(Cp - s(t))}{N(t)} \qquad (7)$$

So that each additional flow, rate path becomes:

$$R_p(t) = R_p(t-d) + \frac{\alpha(C - s(t))}{N(t)} \qquad (8)$$

C as path capacity, $N(t)$ as a number of ongoing flow at time t, α as parameter stability, $R_p(t-d)$ as update rate and $s(t)$ as the measured rate. Rate path $R_p(t)$ is calculated by OpenFlow controller and sent to Switch OpenFlow. Switch OpenFlow then apply the instruction ordered a new controller for each flow or each flow after a certain period *d.*

E. Rerouting

If there is not enough capacity on the primary path or has exceeded a predetermined threshold, then the routing mechanism will divert traffic to another route. Flow will be diverted when the utility of the primary path reaches a value of 0.8 on a path capacity.

$$R_p(t) \geq 0{,}8\, C_p \qquad (9)$$

F. Admission control

Admission control is used to restrict the incoming flow when the network is not able to serve. In this case if the flow is not allowed to enter when added to a new flow, the flow rate will fall below the desired rate of a flow. If the assumption Sending rate allocated to each flow is the same, then the flow rate is the total capacity of a path divided by the number of flows.

$$R\,(t) = \frac{Cp}{N(t)} \qquad (10)$$

A flow entering the network will be rejected if it met the conditions when the flow is received, the flow rate provided is less than the required flow rate ($R\ req$).

$$\frac{Cp}{N(t)} < R\ req \qquad (11)$$

G. Architecture

The system design includes the design of network topology on a data plane and application design SDN. The design of this system is implemented in mininet that allows users to use multiple topologies. This mechanism works by (1) reducing the load node, (2) traffic load balancing and (3) reroutes. To perform these tasks, the system consists of several modules as in figure 3, consists of several processes, namely: The collection of statistical data, calculation of metrics, the selection of best path, traffic distribution: and sending rate determination. We made a simple mechanism, flexible and modular. Consists of five components, namely: collector, metric calculation, rate calculation, and the last flow management.

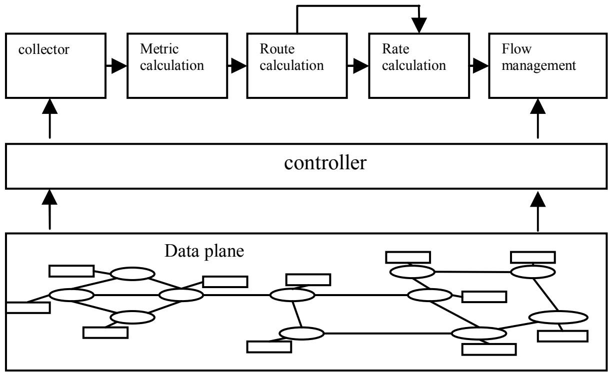

Figure 3: System design

1) Collector: collector module is used to collect information about the condition of the global network, and is responsible for storing all the information. Information can be the network topology, traffic, the condition or state of each device and port, etc.. The controller requested a variety of statistical data from the switch by sending a message FEATURE_REQUEST, and the switch sends a message FEATURE_REPLY contains statistical data requested. The mechanism is described in detail in the specification OpenFlow.

2) Metric calculation: metric calculation module is used for determining link cost parameters that will be used to determine routing and rate and save all the possible paths of each pair of switches.

3) Route calculation: Route calculation module is used to calculate the shortest path and the alternative path between source and destination. Route calculation is done when a message PACKET_IN came to the controller.

4) Rate calculation: Rate calculation module is used to calculate the sending rate. As specified in the OpenFlow specification, any flow of OpenFlow switch contains a set of instructions that run when a packet arrives appropriate entries. One kind of instruction is a meter that directs packets to a specific meter. Each meter has one or more band meter.

5) Flow Management: flow management module is responsible for determining the path of each flow and installs rule / flow entry into datapath or switch. This function is responsible for regulating the flow of efficient by distributing traffic or flow.

IV. Evaluation

To evaluate the proposed scheme we simulate using mininet. We implement the design as an application on the Ryu OpenFlow controller. Simulations using 1 controller and 11 switches, where each switch is logically connected to the controller. Setting parameters based on network topology Abilene [42]. Our evaluation of network performance parameters such as delay, packet loss, and throughput. In this experiment, we apply a single flow and give a background traffic to the network to provide a load on the links and paths.

We observe the simulation results of several different data transfer methods. The method is a single path, multipath without rate adaptation and multipath routing with the rate adaptation. Measurements were made for the UDP data traffic on the link bandwidth of 100 Mbps and set a delay of 1 ms. We use D ITG as a source of traffic. For this scenario, UDP traffic generation, the constant inter-departure time (IDT) in rate 100 PPS and Poisson distributed packet size with = 48 bytes is used. The measurement results in the form of delay, throughput and packet loss as shown in figure 4,5 and 6.

When an incoming flow, the entire bandwidth can be allocated entirely to the flow. When there is subsequent incoming flow, the flow will be allocated a bandwidth of the residual bandwidth path as an equation (10) until when the utility of a path to reach 80%. When the utility path already exceeds 80%, the flow will be diverted to another route. Flow received will be rejected if a newly added flow rate will decrease to below the desired rate. Then the admission control will be applied if the conditions indicated in the equation (11). The simulation results show the throughput with the proposed mechanism to approach the desired value that is equal to 37.98518 kbps. Theoretically, the expected throughput is around 38.4 kbps. The average throughput of the simulation results as shown in Table 1.

TABLE I. AVERAGE THROUGHPUT

No	Data Service	
	Method	*Throughput (kbps)*
1	Single path	36,74179
2	Multipath	37,41376
3	Multipath with rate adaptation	37,98518

The whole of the experimental results showed that the combination of multipath routing methods produces better performance.

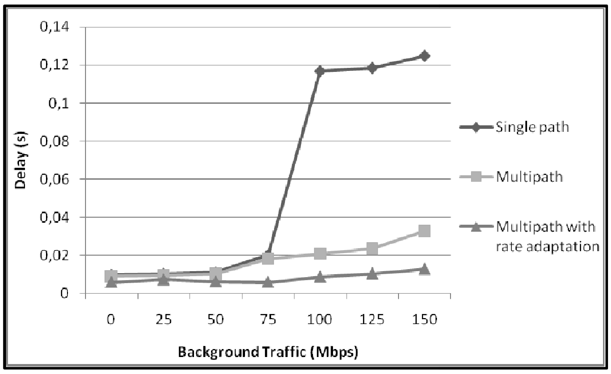

Figure 4: Delay for Data

Seen in Figure 4,5 and 6 when the network load is light experimental results show that the method of single-path, multipath and multipath levels is not too different. Network performance looks significantly better when the utility network is high. It shows that the combination of routing method and rate adaptation can be effectively used to address network congestion. Network load can be divided into various paths. Each path can be allocated a rate corresponding to the bandwidth that is still available.

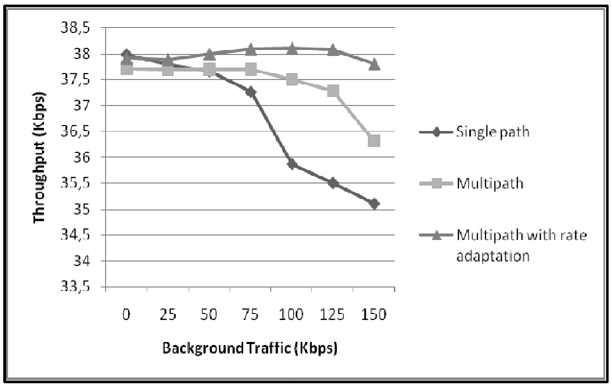

Figure 5: Throughput for Data

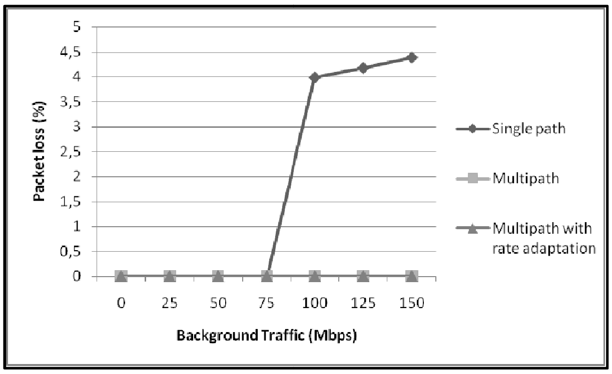

Figure 6: Packet loss for Data

V. CONCLUSION

In this paper, we apply a combination of routing mechanisms and rate adaptation for congestion control in OpenFlow-based software-defined networks. Experimental results show that the combination of multipath routing and rate adaptation produce a better performance than the single path and multipath routing without rate adaptation. On experiments, UDP data transfer at a high bandwidth utility shows that the performance was good and stable. However, the complexity of the multipath routing is higher than the single path. This is due to multipath procedure more complicated than the single path. Besides merging routing and rate adaptation require a longer time. The results listed in this paper are an initial experiment.

For future work, we will observe how much complexity, routing overhead and memory consumption needed. We will conduct experiments to transfer data TCP. We will use different metrics, involving other network parameters are more complex

REFERENCES

[1] Goth, “Software-Defined Networking could shake up more than packets,” IEEE Internet Computing, Vol. 5, no. 4, pp. 6-9, July-August 2011.

[2] Lara, Kolasani, B.Ramamurthy, “Network Innovation using OpenFlow: A Survey,” IEEE Communications Surveys & Tutorials, vol.16, no.1, pp 493 – 512, August 2013.

[3] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling Innovation in Campus Networks,” ACM SIGCOMM Computer Communication Review, vol. 38, no.2, pp. 69-74, April 2008.

[4] Talbot. D, “The Internet is broken,” Technology Review, December 2005-January 2006, http://www.technologyreview.com/article/16356, visited in October 201.

[5] 5. George A. Rovithakis, “End to End Adaptive Congestion Control in TCP/IP Networks,” CRC Press, Print ISBN: 978-1-4398-4057-3, pp.1, 2012.

[6] Van Jacobson and Michael J. Karels, “Congestion Avoidance and Control,” ACM SIGCOMM Computer Communication Review, vol. 18, no. 4, pp. 314-329, August 1988.

[7] K. K. Ramakrishnan and S. Floyd, ”The addition of explicit congestion notification (ECN) to IP, ”Draft IETF (Internet Engineering task force), Sept 2001.

[8] S. Floyd.”TCP and Explicit Congestion Notification,” Comput. Commun. Rev., vol.24 no.5, pp.10–23, October 1994.

[9] Dina Katabi, Mark Handley, and Charlie Rohrs,” Congestion Control for High Bandwidth-Delay Product,” Proceedings of The Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp. 89-102, 2002.

[10] Nandita Dukkipati and Nick McKeown,”Why Flow-Completion Time is the Right Metric for Congestion Control and Why This Means We Need New Algorithms,” ACM SIGCOMM Computer Communication Review, vol. 36, no.1, pp. 59-62, January 2006.

[11] Ershad Sharifahmadian and Shahram Latifi, “A Cognitive Approach for Congestion Control in High Traffic Networks,” 21st International Conference on Systems Engineering, pp.263-266, Las Vegas, August 2011.

[12] Monia Ghobadi, Soheil Hassas Yeganeh, and Yashar Ganjali, “Rethinking End-to-End Congestion Control in Software-Defined Networks,” in proc HotNets-IX, pp. 61-66, Washington, October 2012.

[13] Lu, Yang Xiao, Haifeng Du, “OpenFlow Control for Cooperating AQM Scheme,” Signal Processing (ICSP), IEEE 10th International Conference, pp. 2560 – 2563, Beijing, October 2010.

[14] Kassa Debessay, “ A Network Congestion Control Protocol to More Quickly Finish Flows”, in IDEAL, Agustus 2013, available: http://hdl.handle.net/2142/35905, visit in December 2013.

[15] X. Yang and D. Wetherall, “Source selectable path diversity via routing deflections,” in SIGCOMM, vol.36, pp. 159–170, October 2006.

[16] S. Chen and K. Nahrstedt, “An overview of quality-of-service routing for the next generation high-speed networks: Problems and solutions,” IEEE Network Magazine, vol. 12, no. 6, pp. 64–79, November 1998.

[17] S. De and S. K. Das, “Dynamic Multipath Routing (DMPR): An Approach to Improve Resource Utilization in Networks for Real-Time Traffic,” In Proceedings of MASCOTS’01, pp.23-30, Cincinnati, OH, August 2001.

[18] OpenFlow Switch Consortium and Others. OpenFlow Switch Specification Version 1.3.0. 2012. Available online: https://www.opennetworking.org/images/stories/downloads/sdn-resources/ onf-specifications/openflow/openflow-spec-v1.3.0.pdf (accessed on 2 Apr 2014).

[19] 19. Al-Fares, M., S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat. 2010. “Hedera: dynamic flow scheduling for data center networks”. In Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, 19:1–19:15

[20] P. Narvaez and K. Y. Siu, “Efficient algorithms for multi-path link state routing,” in Proc. ISCOM, pp.320-327, Taiwan, November 1999.

[21] S. Vutukurya and J. Garcia-Luna-Aceves, “MPATH: a loop-free multipath routing algorithm,” Microprocessors and Microsystems, vol. 24, pp. 319–327, October 2000.

[22] S. Vutukury and J. Garcia-Luna-Aceves, “MDVA: A distance-vector multipath routing protocol,” in Proc. INFOCOM, pp.557-564, Anchorage, AK, April 2001.

[23] P. M´erindol, J. Pansiot, and S. Cateloin, “Improving load balancing with multipath routing,” in Proc. ICCCN, pp. 1–8, St. Thomas, US Virgin Islan, August 2008.

[24] Fortz and M. Thorup, “Internet traffic engineering by optimizing OSPF weights,” in Proc. INFOCOM, pp. 519–528, Tel Aviv, March 2000.

[25] R. Banner and A. Orda, “Multipath routing algorithms for congestion,” IEEE/ACM Transactions on Networking, vol. 15, no. 2, pp. 413–424, Apr. 2007.

[26] Z. Wang and J. Crowcroft, “Quality-of-service routing for supporting multimedia applications,” IEEE Journal on Selected Areas in Communications, vol. 14, no. 7, pp. 1228–1234, September 1996.

[27] Q. Ma and P. Steenkiste, “On path selection for traffic with bandwidth guarantees,” in Proc. ICNP, pp.191-202, Atlanta, GA, October 1997.

[28] R. Guerin, S. Kamat, A. Orda, T. Przygienda, and D. Wiliams, “QoS routing mechanisms and ospf extensions,” Global Telecommunications Conference GLOBECOM '97, IEEE, pp. 1903 – 1908, vol.3, Phoenix, AZ, 3-8 November 1997.

[29] Y. Jia, I. Nikolaidis, and P. Gburzynski, “Multiple path QoS routing,” in Proc. ICC, pp. 2583–2587, 2001.

[30] C. Villamizar, “OSPF optimized multipath (OSPF-OMP),” IETF Internet, Feb. 1999, http://tools.ietf.org/html/draft-ietf-ospf-omp-02. visit in December 2013.

[31] C. Villamizar, “MPLS optimized multipath (MPLS-OMP),” IETF Internet Draft, Feb. 1999, http://tools.ietf.org/html/draft-villamizar-mpls-omp-01, visit in December 2013.

[32] S. Nelakuditi, Z.-L. Zhang, and D. H. Du, “On selection of candidate paths for proportional routing,” Computer Networks, vol. 44, pp. 79–102, January 2004.

[33] H. Egilmez, S. Dane, K. Bagci and A. Tekalp, “OpenQoS: An OpenFlow Controller Design for Multimedia Delivery with End-to-End Quality of Service over Software-Defined Networks”, in signal and information Processing Association Annual Summit and Conference (APSIPA ASC) 2012 Asia–Pasific, pp.1-8, Hollywood, LA, December 2012.

[34] Ali Al-Shabibi, Brian Martin, “StepRoute - A MultiRoute Variant Based on CongestionIntervals,” ICN 2011: The Tenth International Conference on Networks, pp.52-56, St. Marteen, Antilles, Netherland, January 2011.

[35] Fern Y. Hunt, Vladimir M,”Dynamic Routing and Congestion Control Through Random Assignment of Routes” Proceedings of the 5th International Conference on Cybernetics and Information Technologies, Systems and Applications: CITSA 2008, pp.161-164, Orlando, July 2008.

[36] Roberto Cominetti, Cristobal Guzman, “ Network congestion control with markovian multipath routing,” NETGCOOP, pp. 1-8, Paris, July 2011.

[37] Jiayue, Ma’ayan Bresler, Mung Chiang and Jennifer Rexton, “Toward Robust Multi-layer traffic engineering: Optimization of Congestion control and routing”, Selected Areas in Communications, IEEE Journal, pp.868-880, vol. 25, June 2007.

[38] Heller, B., S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. Mckeown. 2010. “ElasticTree: saving energy in data center networks”. In Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, 17:1–17:16.

[39] Benson, T., A. Anand, A. Akella, and M. Zhang. 2010a. “The case for fine-grained traffic engineering in data centers”. In Proceedings of the 2010 Internet Network Management Workshop/Workshop on Research Enterprise Networking, 2:1–2:6.

[40] F. Shahrokhi and D. W. Matula, “The maximum concurrent flow problem,” Journal of the ACM (JACM), vol. 37, no. 2, pp. 318–334, 1990.

[41] P.-O. Bauguion, W. Ben-Ameur, and E. Gourdin, “A new model for multicommodity flow problems, and a strongly polynomial algorithm for single-source maximum concurrent flow,” Electronic Notes in Discrete Mathematics, vol. 41, pp. 311–318, 2013

[42] Internet2 Network Infrastructure Technology. Internet2. Retrieved 2015-02-16. [Online]. Available: http://www.internet2.edu/media files/1