2017 4th International Conference on Information Science and Control Engineering

A Survey of Building Efficient P2P Streaming Systems in SDN

Jianwei Zhang (Corresponding Author)*, Chunling Yang†, Huiling Shi*, Lu Wang*
**Shandong Provincial Key Laboratory of Computer Networks*
Shandong Computer Science Center (National Supercomputer Center in Jinan)
Jinan, China
Email: janyway@outlook.com
†*Information Technology Center*
Zhejiang University
Hangzhou, China

***Abstract*—Currently, the contradiction between the rapid growth of network traffic and the limited network resource is plaguing network operators and Internet users. In this paper, we conduct a survey on how to build efficient streaming systems in edge network under SDN (software defined networking) architecture and on how to manage the large traffic in core network caused by file transferring and media streaming applications. We categorize and discuss related studies as P2P (peer-to-peer) optimization at application layer, traffic engineering and QoS routing at network layer, and the most important cross-layer studies, which consist of P2P research in transport and network layers and SDN group applications. Combining the previous research in this field, we propose three potential directions for future research. For each direction, we present a conceptual strategy and its possible implementation in OpenFlow, which can capture the essential features of the SDN architecture and the P2P streaming system.**

***Keywords*-peer-to-peer (P2P); software defined networking (SDN); OpenFlow; streaming; survey**

I. Introduction

Recently, streaming media and large file transfers cover the vast majority of Internet users and consume most of the Internet traffic. Media streaming can be realized by IP multicast, ALM (application layer multicast), CDN and other technologies. Compared with these technologies, P2P (peer-to-peer) has achieved great success due to its simplicity in deployment, excellent performance and strong scalability. Unlike common streaming protocols, such as RTP, RTSP and RTMP, P2P streaming divides the media content into equal-sized blocks for parallel transmission. Most of current P2P streaming systems use BitTorrent as the core, such as Tribler, PPTV, and etc. With the rapid development of mobile Internet, more and more streaming systems begin to support mobile platforms. In addition to P2P clients installed on PC, Tablet, mobile phones and other platforms, a variety of download and streaming services built in web browser and cloud disk also use P2P to get acceleration.

However, the contradiction between the rapid growth of network traffic and the limited network resource is plaguing network operators and Internet users. SDN (software defined networking) and OpenFlow have become hot spots in recent years and are being widely deployed in industry. OpenFlow uses *flow* as the basic unit for traffic optimization, and is fully compatible with traditional IP network. The separation of control plane and data plane can make it better monitor and control the network, and can bring new opportunities for controllability and manageability of the traffic in core network.

Therefore, how to build efficient P2P system in edge networks with the assistance of SDN architecture, and how to manage the large traffic in core network caused by file transferring and media streaming applications have important academic significance and application value. Specifically, this paper has the following contributions.

- We categorize and discuss related studies as P2P optimization at application layer, traffic engineering and QoS routing at network layer, and the most important cross-layer studies, which consist of P2P research in transport and network layers and SDN group applications.
- Combining the previous research in this field, we propose three potential directions for future research. For each direction, we present a conceptual strategy and its possible implementation in OpenFlow, which capture the essential features of the SDN architecture and the P2P streaming system.

II. Survey

When optimizing the P2P streaming system, the theoretical bound for maximum flow rate under different topologies can be obtained by Steiner tree packing or network coding, which is usually tree-based [1]. However, due to the highly dynamic nature of P2P networks, the mainstream optimization works are still based on mesh-based model. Recent research on SDN traffic optimization can be divided into two major categories: traffic engineering and QoS routing. Most of them are based on graph theory and network flow theory, and use maximum link utilization, overall cost, link delay and etc. as the optimization objectives. There are much less studies on the group applications (including IP multicast, ALM, P2P, and etc.) in SDN environment, which

978-1-5386-3013-6/17 $31.00 © 2017 IEEE
DOI 10.1109/ICISCE.2017.35

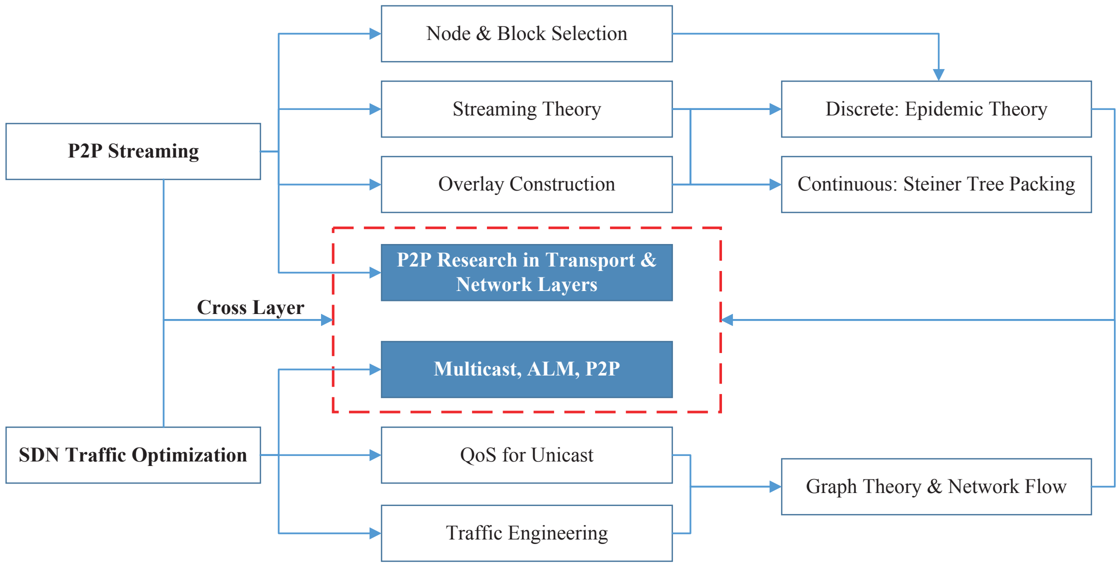

Figure 1. Relationship between related works.

usually have cross-layer characteristics. We summarize the relationship between related works in Fig. 1.

A. Application Layer Optimization for P2P System

With regard to the topology construction algorithm, Dale et al. [2] study the composition of BitTorrent network through real network experiment, i.e.: connection network, interest network, unchoking network and download network. According to the average path length and the cluster coefficient, the connection network, the interest network and the download network exhibit the characteristics of the random graph, and the unchoking network is also characterized by the random graph after the initial stage. The connection network is the superset of other networks. The unchoking network determines the path of data transmission, and has a decisive impact on the scalability and performance of BitTorrent system.

For the node selection and the block selection, Fan et al. [3] study the trade-off between performance and fairness in BitTorrent system. They divide all nodes into N categories according to their bandwidth, and formulate the optimization of the average download time and fairness into linear programming problem. They find that the default strategy of BitTorrent protocol is more focusing on ensuring fairness. In [4], three metrics, i.e.: source load, download time and fairness, of the C/S model, fully cooperative strategy and TFT (Tit-for-Tat) strategy are theoretically derived. The results show that, compared with the fully cooperative strategy with the exponential growth rate, the TFT strategy, which is widely used in P2P systems, only enables linear growth, but has better fairness.

The streaming source can be either real-time media collected by the camera or pre-stored media files. Therefore, the basic theory of live streaming can benefit directly from some conclusions of file sharing. Kumar et al. [5] analyze the equal service and two-class differentiated service when nodes have heterogeneous bandwidth and the initial data contained in each node is zero. For two-class differentiated service, they put forward the concept of forwarding nodes. The expression of completion time for both the equal service and differentiated service are derived.

The maximum flow rate problem is usually addressed by network coding or optimization techniques. If network coding is not allowed, the research is divided into two cases. For structured network, Sengupta et al. [6] divide the maximum flow rate problem into 16 optimization problems according to whether there is multiple session, whether the topology is a complete graph, whether the node degree is limited, and whether there are helper nodes, and gives optimal solutions or approximation algorithms. For unstructured network, Massoulie et al. [8] divide the problem into two parts: edge-capacitated network and node-capacitated network. In former network, the maximum flow rate can be reached by a simple random useful forwarding algorithm (which is a distributed implementation); in the latter one, it is proven in complete graph that, if network coding is not allowed, there is a distributed algorithm can achieve the optimal.

B. SDN Traffic Engineering and QoS Routing for Unicast

In recent years, the promotion of SDN provides new possibilities for the development of media streaming applications. OpenFlow, a standardized southbound interface, is universally recognized by the industry and becomes the focus of academic research. In the modeling of SDN system, Jarschel et al. [9] establish an abstract SDN system, which is composed of a controller and a switch, to assess the packet forwarding rate and blocking probability through queuing theory and simulation. In [10], network calculus is applied to the analysis of SDN controller and switches, and some boundary conditions of SDN deployment, such as extensibility, upper and lower bound for delay, buffer size are taken into account. In OpenFlow network, since topology information, access bandwidth, link capacity can be accurately obtained, the monitoring methods for real-time traffic, delay and other dynamic information have been gradually developed [15], [16], and some open source systems begin to take shape.

Research on traffic engineering has undergone three main stages: equal-cost multi-path routing (ECMP), MPLS tunnel, and SDN [21]. In the traditional OSPF routing method, the link weight is usually set to be inversely proportional to the link capacity or proportional to the physical length of the link, which may cause network congestion. To improve network performance, the OSPF weights should be optimized to balance the flows. Around this issue, there are a number of improvements based on OSPF. Agarwal et al. [11] first study how to place SDN switches in a given network topology in an efficient way and how the number of SDN switches will impact on the network performance. In addition, they define the traffic through the SDN switches as controllable traffic, and the traffic through traditional switches/routers as uncontrollable traffic, and propose an approximate algorithm of polynomial time to solve the dynamic routing problem. But they assume that the link weights are all equal. In [12], the assumption is relaxed, and both the weight setting of links and the splitting ratio of flows are optimized. In the SDN/OSPF hybrid network, the proposed local search heuristic algorithm can reduce the maximum link utilization. Floyd et al. [13] demonstrate that hierarchical link bandwidth sharing in traditional networks can provide a high degree of flexibility for the use of new types of applications and network protocols, and their conclusions are still applicable for SDN.

Karakus et al. [7] review related studies where QoS can benefit from the SDN architecture. The studies are divided into several classes, such as multimedia flows routing, inter-domain routing, resource reservation, queue management and scheduling, QoE-aware and QoS-centric routing, network monitoring. In [20], an OpenQoS controller based on OpenFlow is proposed, which can provide end-to-end QoS support and can realize high quality HTTP adaptive streaming service with different bit rates in dynamic network environment. In [14], an end-to-end QoS routing strategy with delay and jitter as the main metrics in SDN is proposed for high-quality video transmission. In SVC stream, the base layer and the enhancement layer use routing strategies with different priorities. The base layer turns to the OpenFlow architecture for QoS routing, while the enhancement layer uses the traditional best-effort routing. In [22], an application-aware method based on SDN is proposed. Taking YouTube streaming as an example, the experiments compare the QoE and major overhead of deep packet inspection, directly inputting application information to the controller, the round-robin path selection, and the load balancing path selection.

C. P2P Research in Transport & Network Layers and SDN Group Applications

Existing research in this field mainly focuses on verifying via simulation how system performance changes under different transport layer protocols or different parameter configurations. As the packet-level P2P simulation involves a large number of data block exchange, which results in very large computation complexity, only a small part of current research studies congestion control in this way. For example, Testa et al. [17] study a low-priority congestion control protocol LEDBAT through NS2 simulation, which is recently introduced to BitTorrent standard. In addition, most P2P simulators are designed at flow-level. For example, in [18], network layer and traffic model are added in P2P simulation, and the link bandwidth allocation and release mechanism of application layer connection is designed.

As to SDN group applications, Niyonteze et al. [25] propose a content-oriented IPTV content distribution system, which can be implemented through the OpenFlow group table mechanism. In [26], a multicast system SDM^{2}Cast is designed to provide both the extensibility and flexibility of media flow management. Its effectiveness is verified combined with SVC video streams and heterogeneous end nodes. Georgopoulos et al. [23] propose an OpenCache system, including an OpenCache controller and an OpenCache node, to provide cache service for the VoD streams. The global calculation of the OpenCache controller is used to cache the video resources at the nearest OpenCache nodes from the end users to reduce repeated unicast. Li et al. [30] propose a CacheFlow scheme which requires the OpenFlow switches to provide the caching function. The effects of different cache placement strategies are verified by simulation. In [24], a cross-layer application RASP is designed for P2P live streaming in ISP-managed OpenFlow controllers, which includes virtual peer application, software defined multicast (SDM) and API interface provided in application layer. The main idea is to use some ISP-managed nodes as virtual nodes, and push the live stream to end nodes in access network in the form of multicast tree according to the optimization result calculated by the controller. In the core network, the network layer multicast is implemented in OpenFlow switches to reduce the repeated traffic. In [27], an application layer routing method based on packet request redirection is proposed. The packet sent to the original content server is redirected to other end nodes that already have the content with assistance of the OpenFlow architecture, so that the end nodes can bypass the content server to directly exchange data in P2P mode. In content-based network [28], the content redirection is achieved through anycast, which can increase the number of serviceable nodes and reduce the startup time of the nodes to join the P2P network. In [29], SDrN (Software Defined residential Network) is proposed to provide P2P multicast service in ISP-managed SDN access network. Through the OpenFlow architecture, the bandwidth of newly joined nodes and nodes with nearest topology distance is adjusted so that the QoS adjustment can be achieved at the network layer. In Mininet environment [19], [31], a ring-shaped topology at network layer, which is made to periodically change by the SDN controller, is used to observe how the node selection strategy of the BitTorrent system is affected. The results show that the network layer strategy of SDN is not consistent with the built-in strategy of BitTorrent system. They both have passive reactions to each other's strategy.

III. Potential Directions for Future Research

A. Traffic Scheduling Strategy

Fig. 2 briefly shows the traffic scheduling strategy. The traffic demands of nodes A and C are denoted by D(A) and D(C) respectively. Assume that the shortest path is ab, and if D(A) exceeds the maximum capacity of link ab, congestion will occur. So we can use the links ac and cb to forward part of the traffic generated by A. The traffic scheduling process may be limited by the capacity of each link.

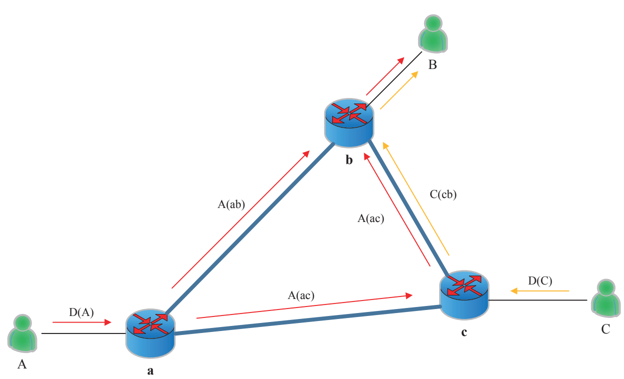

Figure 2. Traffic scheduling strategy.

A(ab) and A(ac) represent the flows on links ab and ac sent by node A, respectively. C(cb) is the traffic on link cb sent by node C. The flows targeted to the same node may aggregate in the same link and cause the link to become congested and reducing network performance [11], [12]. Because the SDN controller has the global topology view and knows the real-time load of each link, the SDN controller can dynamically select the independent output ports for each flow and distribute the traffic to different links. This can be regarded as the multi-commodity flow problem in network flow theory. In existing technologies, by configuring the forwarding devices as the bond mode (enabling the LACP protocol), port and interface can be used as a one-to-many relationship to achieve dynamic link aggregation between directly connected switches, which can provide better flexibility and controllability. For some delay-insensitive applications, splitting a stream to different paths has no serious impact; while for delay-sensitive applications, it is necessary to ensure a stream is transmitted along the same path. Therefore, the traffic scheduling strategy for different application types needs to be further studied.

B. Link Bandwidth Allocation Strategy

Fig. 3 briefly shows the link bandwidth allocation strategy. Nodes A and C are sending data to node B at the same time. P2P connections AB and CB may compete for available bandwidth of link bB. According to the built-in incentive mechanism in BitTorrent system, the amount of data transmitted on AB and CB directly affects the decision of node B as to whether node A or node C has greater contribution to it. If the amount of data transmitted on AB is too small for a period of time, node B will reduce the probability of transmitting data to node A, or even disconnect node A and try to connect to other nodes instead.

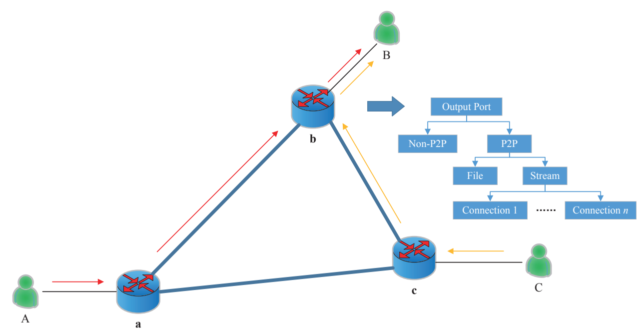

Figure 3. Link bandwidth allocation strategy.

OpenFlow protocol is able to define any combination of the packet header fields as a flow, and supports setting queue ID for any packet in the flow. To realize the link bandwidth allocation strategy, the SDN controller identifies each flow and sends flow tables to the SDN switches, and then the output ports execute scheduling. The QoS function can be realized by combining Open vSwitch and the flow control module in Linux kernel which implements the hierarchical token bucket. It can provide bandwidth rate limiting and hierarchical flow classification for each flow according to different QoS policies [13]. Using bandwidth rate limiting and hierarchical traffic classification, we can achieve a number of specific strategies, such as limiting the foreground traffic to a certain percentage; scheduling flows via averaging strategy or max-min fair strategy, and so on. How to allocate bandwidth for different applications and connections on the SDN switch ports in the global network view, and how to allocate and release bandwidth at the link level in experiments will be the focus of further research.

C. QoS Routing Strategy

Fig. 4 briefly illustrates the QoS routing strategy. Node A is sending SVC data to node B. Traditional P2P system uses exactly the same route on a connection. In the SDN environment, the base layer can be transmitted along the route AacbB, which is dynamically calculated by the QoS routing algorithm in the SDN controller. Meanwhile, the enhancement layer is transmitted along AabB in the shortest path first manner. The basic idea is to give precedence to the route with largest available bandwidth rather than the shortest path [14]. When there are multiple enhancement layers, each enhancement layer can also adopt a specified QoS routing policy. Obviously, since node B receives SVC data of different layers at different rates, its block selection strategy will be affected.

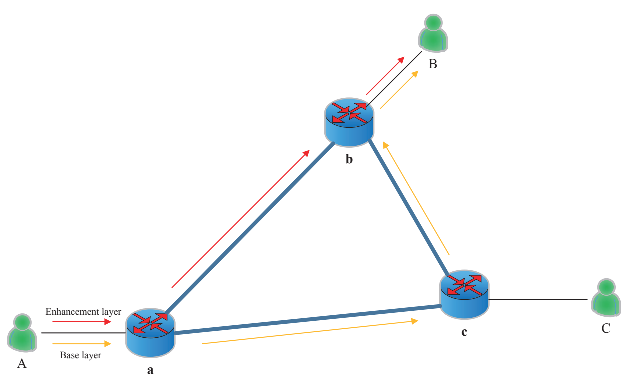

Figure 4. QoS routing strategy.

Compared with the additive QoS metrics such as delay, bandwidth-based QoS routing only needs to find the bottleneck bandwidth along the links on the overlay connection. Different from RTP and other unicast streaming protocols, P2P media streaming usually uses media block as the basic unit, and the QoE of end users comes directly from the block transferring efficiency. Delay, jitter and other metrics have indirect impacts on the video playback fluency through their impacts on block transmission efficiency. Further research may consider video coding modes, and may derive cost function for QoS routing strategy, using bandwidth as the primary parameter, packet loss ratio, delay and jitter as the subordinate parameters. On the other hand, existing research on streaming efficiency generally uses startup delay and playback probability as the main performance metrics. PSNR (peak signal to noise ratio) can also be introduced to carry out the evaluation, which involves the mapping between the streaming media block and video coding unit. In addition, whether the non-equal splitting of streaming media block is feasible is also worth studying .

IV. Conclusion

In this paper, we conduct a survey on how to build efficient streaming systems in edge network under SDN architecture and on how to manage the large traffic in core network caused by file transferring and media streaming applications.

We categorize and discuss related studies as P2P optimization at application layer, traffic engineering and QoS routing at network layer, and the most important cross-layer studies, which consist of P2P research in transport and network layers and SDN group applications. Combining the previous research in this field, we propose three potential directions for future research. For each direction, we present a conceptual strategy and its possible implementation in OpenFlow, which capture the essential features of the SDN architecture and the P2P streaming system.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant No. 61472230, the Shandong Provincial Natural Science Foundation of China under Grant No. ZR2015YL021, ZR2015YL019, ZR2014YL042 and ZR2015FL025, and the Youth Science Funds of Shandong Academy of Sciences under Grant No. 2016QN004.

References

[1] Y. Wu, K. Jain, and S. Y. Kung, "A unification of network coding and tree-packing (routing) theorems," IEEE Transactions on Information Theory, vol. 52, no. 6, 2006, pp. 2398–2409.

[2] C. Dale, J. Liu, J. Peters, and B. Li, "Evolution and enhancement of bittorrent network topologies," 2008 16th IEEE International Workshop on Quality of Service (IWQoS). IEEE, 2008, pp. 1–10.

[3] B. Fan, J. Lui, and D. M. Chiu, "The design trade-offs of BitTorrent-like file sharing protocols," IEEE/ACM Transactions on Networking, vol. 17, no. 2, 2009, pp. 365–376.

[4] D. DeFigueiredo, B. Venkatachalam, and S. F. Wu, "Bounds on the performance of P2P networks using tit-for-tat strategies," 2007 7th IEEE International Conference on Peer-to-Peer Computing. IEEE, 2007, pp. 11–18.

[5] R. Kumar and K. Ross, "Optimal peer-assisted file distribution: single and multi-class problems," 2006 IEEE Workshop on Hot Topics in Web Systems and Technologies (HOTWEB'06). 2006.

[6] S. Sengupta, S. Liu, M. Chen, M. Chiang, J. Li, and P. A . Chou, "Peer-to-peer streaming capacity," IEEE Transactions on Information Theory, vol. 57, no. 8, 2011, pp. 5072–5087.

[7] M. Karakus and A. Durresi, "Quality of service (QoS) in software defined networking (SDN): a survey," Journal of Network and Computer Applications, vol. 80, 2017, pp. 200-218.

[8] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized decentralized broadcasting algorithms," INFOCOM 2007. IEEE, 2007, pp. 1073–1081.

[9] M. Jarschel, S. Oechsner, D. Schlosser, R. Pries, S. Goll, and P. Tran-Gia, "Modeling and performance evaluation of an OpenFlow architecture," 23rd International Teletraffic Congress, 2011, pp. 1–7.

[10] S. Azodolmolky, R.Nejabati, M. Pazouki, P. Wieder, R. Yahyapour, and D. Simeonidou, "An analytical model for software defined networking: a network calculus-based approach," 2013 IEEE Global Communications Conference (GLOBECOM). IEEE, 2013, pp. 1397–1402.

[11] S. Agarwal, M. Kodialam, and T.V. Lakshman, "Traffic engineering in software defined networks," INFOCOM 2013. IEEE, 2013, pp. 2211–2219.

[12] Y. Guo, Z. Wang, X. Yin, X. Shi, and J. Wu, "Traffic engineering in SDN/OSPF hybrid network," 2014 22nd IEEE International Conference on Network Protocols (ICNP). IEEE, 2014, pp. 563–568.

[13] S. Floyd and V. Jacobson, "Link-sharing and resource management models for packet networks," IEEE/ACM Transactions on Networking, vol. 3, no. 4, 1995, pp. 365–386.

[14] H. E. Egilmez, S. Civanlar, and A. M. Tekalp, "An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks," IEEE Transactions on Multimedia, vol. 15, no. 3, 2013, pp. 710–715.

[15] K. Phemius and M. Bouet, "Monitoring latency with OpenFlow," 2013 9th International Conference on Network and Service Management. IEEE, 2013, pp. 122–125.

[16] D. Raumer, L. Schwaighofer, and G. Carle, "MonSamp: a distributed SDN application for QoS monitoring," 2014 Federated Conference on Computer Science and Information Systems. IEEE, 2014, pp. 961–968.

[17] C. Testa and D. Rossi, "Delay-based congestion control: flow vs. BitTorrent swarm perspectives," Computer Networks, 2014, vol. 60, pp. 115–128.

[18] W. Yang and N. Abu-Ghazaleh, "GPS: a general peer-to-peer simulator and its use for modeling BitTorrent," 2005 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. IEEE, 2005, pp. 425–432.

[19] D. Vicino, C. H. Lung, G. Wainer, and O. Dalle, "Evaluating the impact of software-defined networks' reactive routing on Bittorrent performance," Procedia Computer Science, 2014, vol. 34, pp. 668–673.

[20] H. E. Egilmez, S. T. Dane, K. T. Bagci, and A. M . Tekalp, "OpenQoS: an openflow controller design for multimedia delivery with end-to-end quality of service over software-defined networks," 2012 Asia-Pacific Signal & Information Processing Association Annual Summit and Conference. IEEE, 2012, pp. 1–8.

[21] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "A roadmap for traffic engineering in SDN-OpenFlow networks," Computer Networks, 2014, vol. 71, pp. 1–30.

[22] M. Jarschel, F. Wamser, T. Hohn, T. Zinner, and P. Tran-Gia, "SDN-based application-aware networking on the example of YouTube video streaming," 2013 2nd European Workshop on Software Defined Networks. IEEE, 2013, pp. 87–92.

[23] P. Georgopoulos, M. Broadbent, B. Plattner, and N. Race, "Cache as a service: leveraging sdn to efficiently and transparently support video-on-demand on the last mile," 2014 23rd IEEE International Conference on Computer Communication and Networks (ICCCN). IEEE, 2014, pp. 1–9.

[24] J. Ruckert, J. Blendin, and D. Hausheer, "ASP: using OpenFlow to push overlay streams into the underlay," 2013 13th IEEE International Conference on Peer-to-Peer Computing. IEEE, 2013, pp. 1–2.

[25] B. Niyonteze and J. W. K. Hong, "Provider-side VoD content delivery using OpenFlow multicast," KNOM Review, vol. 17, no. 1, 2014.

[26] J. Yang, E. Yang, Y. Ran, and S. Chen, "SDM^{2}Cast: an OpenFlow-based, software-defined scalable multimedia multicast streaming framework," IEEE Internet Computing, vol. 19, no. 4, 2015, pp. 36–44.

[27] O. M. M. Othman and K. Okamura, "Design and implementation of application based routing using OpenFlow," 5th International Conference on Future Internet Technologies. ACM, 2010, pp. 60–67.

[28] O. M. M. Othman and K. Okamura, "On demand content anycasting to enhance content server using P2P network," IEICE Transactions on Information and Systems, 2012, vol. 95, no. 2, pp. 514–522.

[29] I. Trajkovska, P. Aeschlimann, C. Marti, T. M. Bohnert, and J. Salvachúa, "SDN enabled QoS provision for online streaming services in residential ISP networks," 2014 IEEE International Conference on Consumer Electronics-Taiwan. IEEE, 2014, pp. 33–34.

[30] T. Li, N. Van Vorst, R. Rong, and J. Liu, "Simulation studies of OpenFlow-based in-network caching strategies," 15th Communications and Networking Simulation Symposium. Society for Computer Simulation International, 2012, pp. 12.

[31] M. K. Guraya, R. S. Bajwa, D. Vicino, and C. H. Lung, "The assessment of BitTorrent's performance using SDN in a mesh topology," 2015 6th IEEE International Conference on the Network of the Future (NOF). IEEE, 2015, pp. 1–3.