LEVERAGING SDN TO IMPROVE THE PERFORMANCE OF MULTICAST-ENABLED IPTV DISTRIBUTION SYSTEMS

Roberto Canonico and Simon Pietro Romano

ABSTRACT

Distribution of live multimedia streams to a high number of end users through an ISP network may be efficiently implemented through IP multicasting. Multicast traffic engineering with traditional decentralized approaches, however, is still a challenging problem for network operators. In this article we present a centralized approach to management of source-specific multicast traffic flows relying on software defined networking. Our approach is suitable for hybrid IP/SDN networks, where network nodes behave as traditional IP routers by default, but whose packet forwarding behavior may be customized for specific flows thanks to an OpenFlow-enabled control plane. We also describe a traffic management system that we implemented as a proof of concept, which can be used by an ISP to compute multicast distribution trees for a given set of channels to be distributed to their customers.

INTRODUCTION

A recent Cisco VNI report estimates that IP video will represent 82 percent of all consumer Internet traffic globally by 2021, up from 67 percent in 2014. In particular, live video is expected to grow 15-fold from 2016 to 2021 [1]. Distribution of live multimedia streams to a high number of end users through the Internet is a big challenge for both network operators and IPTV content providers.

Engineering the distribution infrastructure in a way that makes it able to accommodate a very large number of *peak concurrent plays* requires more efficient transmission techniques within the network. As of today, over-the-top (OTT) players adopt the traditional unicast service model that can seamlessly reach any end user connected to the Internet at the cost of huge investments to deploy powerful servers to concurrently manage large numbers of high-bandwidth traffic flows. Significant cost savings could be achieved if a universal inter-domain multicast IP delivery service were available for the global Internet. Unfortunately, after many years of research, this is still not the case.

Within single Internet service providers' (ISPs') networks, instead, multicast IP is more widely supported. Some ISPs have recently started to use multicast IP to distribute multimedia streams to their customers. BT, for instance, uses multicast IP to distribute the HD BT Sport channel to YouView boxes connected via fiber. Once multicast is enabled within an ISP network, multicast distribution from well-known sources to a potentially large subset of customers might also be provided as a service to third-party content providers. As an example, Sichuan Mobile, China's largest commercial OTT office, reformed in 2015 its content delivery network (CDN) architecture to ensure busy-hour service quality by introducing OTT multicast technologies to reduce backbone network traffic.

Nonetheless, performing efficient traffic engineering of multicast traffic is still a difficult task for ISPs with existing multicast routing protocols.

Protocol Independent Multicast Sparse Mode (PIM-SM, RFC 4601), which is the most widely adopted multicast routing protocol in the Internet, builds a shortest-path tree (SPT) to connect a source and destinations. Traffic engineering with PIM-SM is difficult due to the fact that it only computes SPTs, so it misses the opportunity to reduce bandwidth consumption by sharing the most common edges toward different destinations.

The emerging software defined networking (SDN) paradigm has recently been proposed to perform centralized traffic engineering in ISP networks [2]. The adoption of SDN in transport networks (T-SDN) is described in [3], where the authors describe the high-level reference T-SDN architecture emerging from early experience, standards, and related industry activities. So far, only a few papers have dealt with the problem of traffic engineering of multicast traffic in SDN networks (e.g., [4]). Since a sharp transition of ISP networks to a pure SDN model is not viable, it is reasonable to expect that SDN will coexist with traditional IP routing and management [5]. This coexistence is implemented in so-called hybrid IP/SDN networks. Traffic engineering methods to be adopted in hybrid IP/SDN networks have been investigated only recently [6].

In this work we present a software system that allows ISP operators to manage source-specific multicast distribution trees by assigning a cost to each link and computing the resulting minimum spanning tree. By relying on an OpenFlow-enabled control plane in network nodes, the management application also interacts with an OpenFlow controller to enforce the flow rules that are needed to forward traffic according to

Digital Object Identifier:
10.1109/MCOMSTD.2017.1700036

The authors are with the University of Napoli.

the computed tree.

In [7] we presented the Virtual Puppet Master (VPM), an SDN-based system for creating and managing a virtualized infrastructure for distribution of live streams on top of an SDN-enabled network. Our work assumed that all network nodes were SDN switches and that they were also able to host KVM-based virtual machines which could be used for multimedia transcoding and adaptation. VPM was able to instantiate and live migrate VMs to reconfigure the distribution infrastructure at runtime. In this work we have re-engineered the VPM system to manage a multicast distribution system operating on top of a hybrid IP/SDN network, in which network nodes are IP routers that have been extended with SDN functionality to operate customizable forwarding decisions on specific flows.

The rest of the article illustrates how the software system we developed can be used to manage multicast traffic flows in a typical IPTV distribution system. We also present a prototype of the system that interacts with the Floodlight SDN controller to manage a set of Linux-based IP routers configured to behave as hybrid IP/SDN nodes in combination with Open vSwitch.

ARCHITECTURE OF AN IPTV SYSTEM

IPTV distribution systems are typically implemented according to the architecture depicted in Fig. 1, which is adapted from [8, 9]. Live video feeds are received from terrestrial and/or satellite digital video broadcast (DVB) systems and injected into the ISP's core network from a central super hub office (SHO). Other regional contents may also be distributed from peripheral video hub offices (VHOs). Actual delivery to end users is performed in the ISP's access network, from video serving offices (VSOs) containing the aggregation routers connected to subscribers' homes through a local digital subscriber line access multiplexer (DSLAM).

Video content is encoded using an encoding standard (e.g., H.264) and delivered to set-top boxes by means of the Real-Time Transport Protocol (RTP) on top of UDP.

Each TV channel is assigned to a unique multicast group.

A typical IPTV system delivers hundreds of both standard-definition (SD) and high-definition (HD) channels (the former producing an average traffic between 1.5 and 3 Mb/s, the latter between 6 and 10 Mb/s). The resulting aggregate bandwidth is hence on the order of multiple gigabits per second.

For IPTV two different multicast service models may be adopted [10]: any source multicast (ASM) and source-specific multicast (SSM). With SSM the network layer service is a "channel," identified by an SSM destination IP multicast address G and a source IP address S [11]. The Internet Assigned Numbers Authority (IANA) has assigned the IPv4 address range 232/8 and the IPv6 address range FF3x::/96 as multicast addresses reserved for SSM services.

At the edges of the network, customer equipments produce Internet Group Multicast Protocol (IGMP) signaling when a channel is selected (JOIN message) or has left (LEAVE message). When the SSM service model is adopted, receivers use IGMPv3 messages to explicitly join a channel (i.e., a combination of a source S and a multicast destination group G). In a traditional multicast IP network, routers create and maintain source-specific forwarding rules for each (S,G) channel. One of the advantages of the SSM service model is security: ISPs may allow only a few well-known hosts to act as channel sources [10].

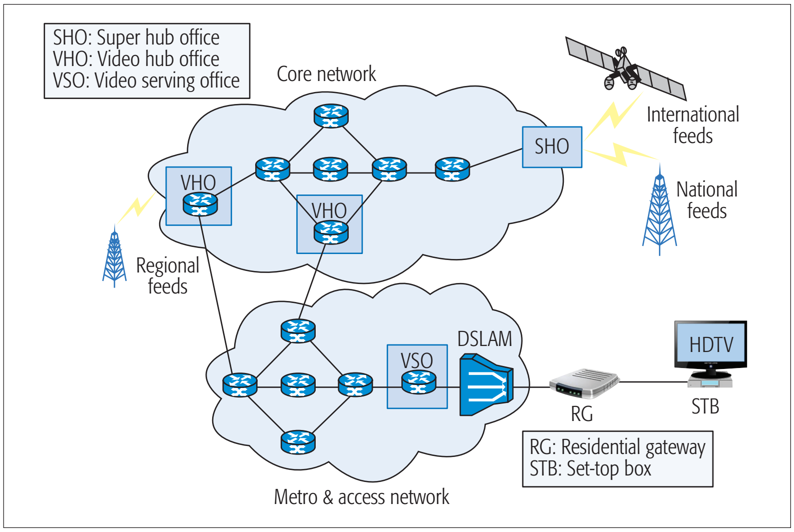

FIGURE 1. Architecture of an IPTV system.

IPTV AND SDN RELATED STANDARDIZATION ACTIVITIES

The need for efficient and flexible management platforms for content distribution infrastructures has been recognized and addressed by all the standardization fora operating in the field of telecommunications. The International Telecommunication Union — Telecommunication Standardization Sector (ITU-T) addressed the use of IP multicast for IPTV content delivery in Supplement 16 of the Y-series Recommendations [10]. An architecture using IP multicast for IPTV distribution of adaptive bit rate (ABR) video streams is described in a CableLabs technical report (OC-TR-IP-MULTI-ARCH) [12].

The Streaming Video Alliance (SVA), an industry forum composed of leading companies from the online video ecosystem, has recently announced a multicast ABR proof of concept (PoC) [13]. This PoC is intended to address 4K streaming to OTT devices over a multicast-enabled IPTV network.

The recent advent of SDN has raised the need for defining standard control plane protocols [14]. OpenFlow is defined by the Open Networking Foundation (ONF). In many contexts, the use of SDN is combined with network functions virtualization (NFV), that is, the ability to implement network functions by means of virtualized software entities. SDN is an essential enabler for NFV as it provides the flexibility of network infrastructure needed by NFV.

ITU-T Study Group SG13 is working on questions covering cloud, SDN, and NFV. In particular, they address the use of these technologies in next

The SDM2 application allows the network administrator to assign a positive cost value to each network link. By computing a minimum cost spanning tree by means of the Kruskal algorithm, the administrator is able to deliver multicast traffic to all VSOs by moving it away from links that may be congested by unicast traffic.

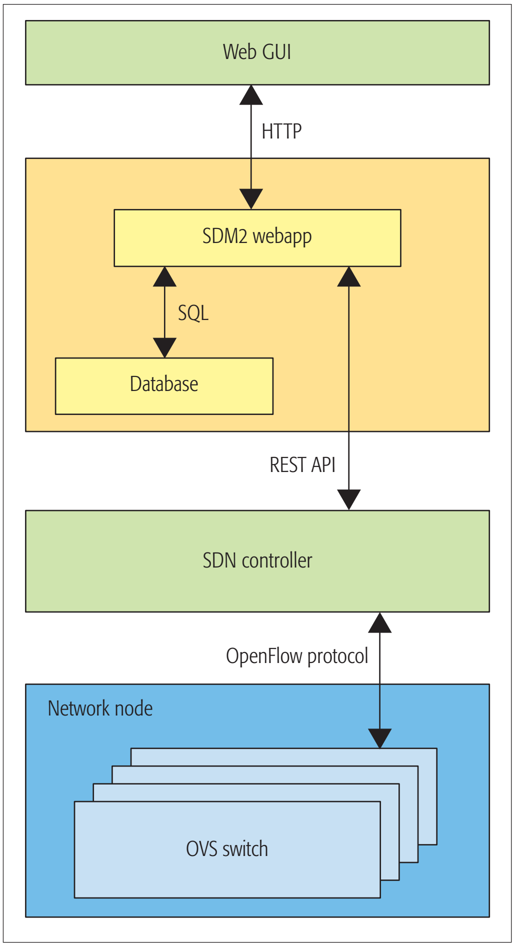

FIGURE 2. Architecture of the SDM2 software system.

generation networks. ITU-T Recommendation Y.3300, "Framework of Software-Defined Networking," provides a basis for the further study of SDN by providing its definition, objectives, capabilities, requirements, and high-level architecture. ITU-T Study Group SG16, among its goals, aims to define standard signaling protocols and application programming interfaces (APIs) for virtual CDNs (vCDNs). To harmonize the SDN-related activities, ITU-T has also established a specific Joint Coordination Activity on SDN (JCA-SDN).

The European Telecommunications Standards Institute (ETSI) has identified the necessity for "a consistent management and orchestration architecture" as one of the challenges to be addressed for successfully implementing NFV. In [15], ETSI envisions dynamic creation of on-demand CDNs as a relevant use case for NFV.

SYSTEM ARCHITECTURE

The system we developed is intended to manage the distribution of multicast traffic from SHOs down to peripheral VSOs in a hybrid IP/SDN network. To this purpose, a source-specific multicast distribution tree is created and enforced in the network by properly configuring OpenFlow switches located within network nodes. If all the network nodes are OpenFlow-compliant, no multicast routing needs to be enabled in network nodes. In this manner, the ISP has complete centralized control over the routes set up for multicast traffic.

Nodes are basically IP routers, whose routing tables are managed by means of an inter-domain routing protocol, such as Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS). Since multicast routing protocols are not enabled in routers, the network is not natively able to deliver multicast traffic. Multicast traffic, instead, is managed thanks to a set of OpenFlow flow rules instantiated in network nodes under control of a centralized SDN controller. Hence, we follow the *class-based hybrid SDN* model defined in [5]: legacy distributed IP routing and centralized SDN control coexist in all network nodes. In our scenario, we adopt SDN only to handle multicast traffic.

The system is composed of three components:

- Hybrid IP/SDN network nodes
- An SDN controller that manages the nodes of the network
- A web-based application (SDM2) that retrieves the network topology from the controller and provides the network administrator with a web graphical user interface (GUI) through which he/she can compute and enforce multiple multicast distribution trees.

The SDM2 application allows the network administrator to assign a positive cost value to each network link (the default value being 1). By computing a minimum cost spanning tree by means of the Kruskal algorithm, the administrator is able to deliver multicast traffic to all VSOs by moving it away from links that may be congested by unicast traffic.

Figure 2 shows the architecture of the software system. The SDM2 application interacts with the SDN controller through a controller-specific northbound API. In our prototype, we used the Floodlight controller, and interaction is based on its REST API. The next section illustrates how we implemented a hybrid IP/SDN network node in Linux.

NODE ARCHITECTURE

Our goal is to extend the behavior of an IP router with SDN capability. The starting point was a plain Linux system, equipped with multiple network interfaces and capable of forwarding packets from one interface to another (by setting to 1 the `net.ipv4.ip_forward` Linux kernel parameter).

By running the Quagga routing software suite, the node behaves as a traditional IP router. In our prototype, we configured Quagga to enable the OSPF routing daemon for IPv4 traffic. Due to the limited size of the testbed, all the nodes were configured to belong to the same OSPF area.

Figure 3 shows a node equipped with four physical network interface cards (NICs): `p1p1`, `p1p2`, `p2p1`, and `p2p2`. Actually, the node is also equipped with a fifth NIC (`em1`). We refer to this NIC as the *management interface*. All the nodes of a network have the management interface connected to a physically separated management network that is used to connect the nodes to a single centralized OpenFlow controller. We configure each node by creating an Open vSwitch software switch for each of the physical NICs of the node. In Fig. 3, these software switches are called `br-p1p1`, `br-p1p2`, `br-p2p1`, and `br-p2p2`, respectively. Internally, all these switches are connected to an internal software switch called `br-int`.

Such an internal architecture of nodes makes it possible to specifically select which traffic flows need to be processed by the SDN logic. As a general rule, all traffic flows that are not specifically selected by the administrator are subject to the default packet processing behavior of an IP router. This is obtained by configuring all the edge Open vSwitch bridges with OpenFlow rules that, by default, direct both incoming and outgoing traffic toward the LOCAL port, which allows these packets to be processed by the Linux kernel as usual, unless a higher-priority flow rule diverts them to the internal `br-int` bridge. Such processing, in particular, is reserved for multicast traffic, whose routing is completely under control of the SDM2 application.

PROTOTYPE TESTING

Figure 4 shows a simple testbed we created to test the SDM2 application. It consists of four HP Proliant N54L micro servers, equipped with an AMD Turion II Neo N54L dual core processor and 8 GB RAM. Three of the nodes (node-1, node-2, and node-3 in Fig. 4) run Ubuntu Linux 14.04 LTS and are configured to behave as IP routers, with IP forwarding enabled. The fourth node runs VMware vSphere ESXi 5.5 and hosts two VMs: a "pfSense VM," acting as NAT router for the whole testbed, and a "Network Controller VM" running Ubuntu Linux 14.04 LTS. The Controller VM hosts both Floodlight, an Apache licensed Java-based OpenFlow controller, and the SDM2 web application running in the Apache Tomcat web container.

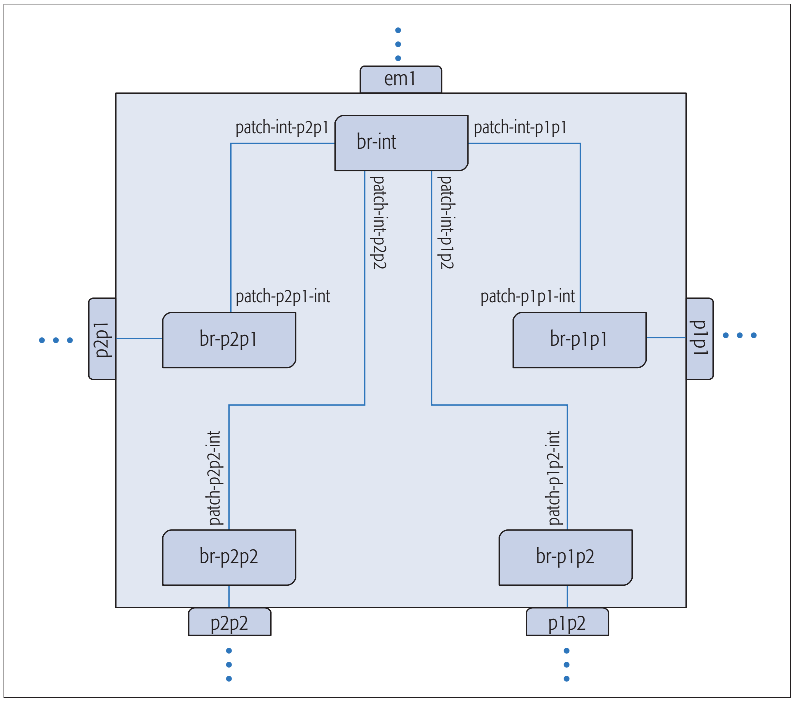

FIGURE 3. Architecture of a hybrid IP/SDN node.

The three nodes acting as routers are connected in a ring topology, as depicted in Fig. 4. Node-1 is also connected to two end systems: SRC and RCV. Node-3, on the other side of the testbed, is connected to an IP set-top box (STB), which is natively able to receive and display H.264 video streams transmitted in RTP. For the purpose of testing the ability of the system to deliver multicast streams to end systems, we have developed some simple GStreamer scripts: one for transmitting an H.264 encoded video (with AAC audio),

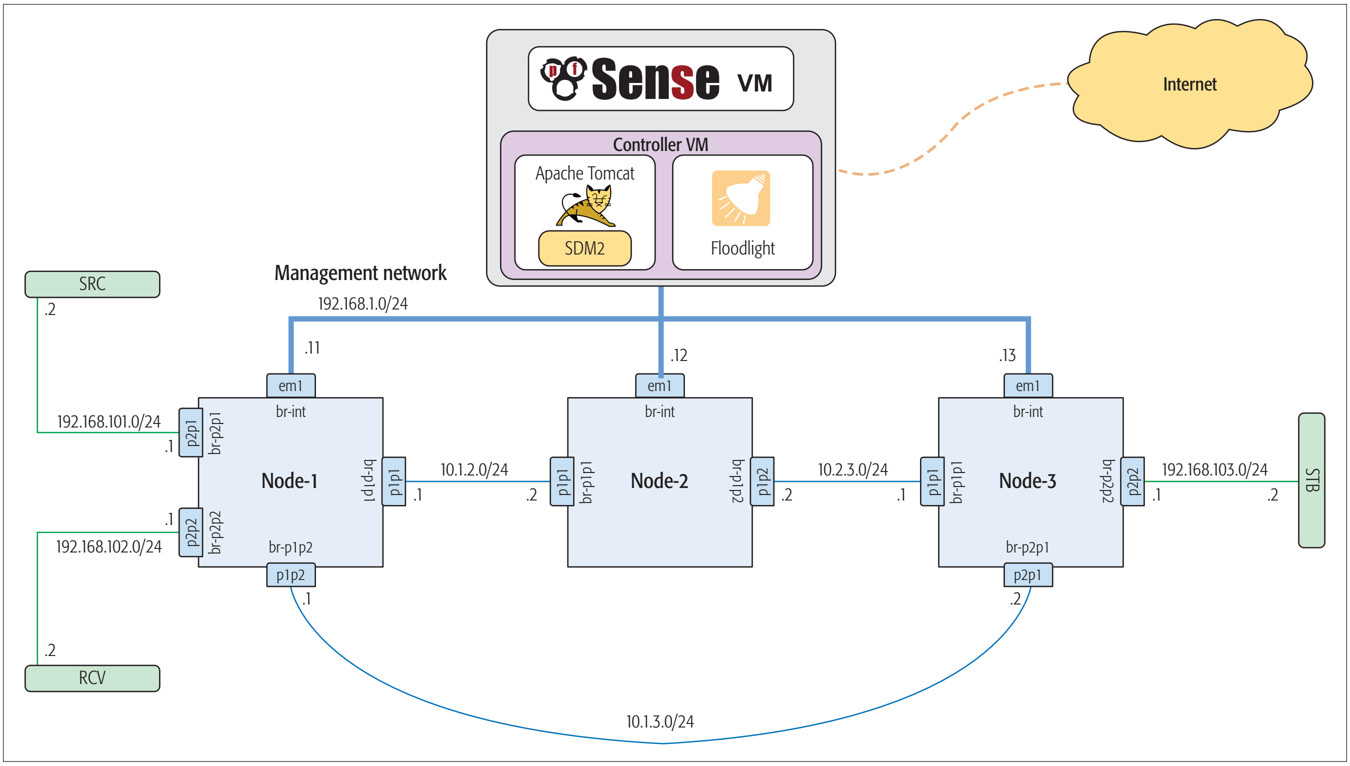

FIGURE 4. Our testbed configuration.

We believe that this approach is particularly well suited for managing IPTV infrastructures in ISP networks, as it allows to implement centralized traffic engineering of multicast traffic. We are currently working toward the implementation of more complex traffic engineering procedures in larger scale scenarios.

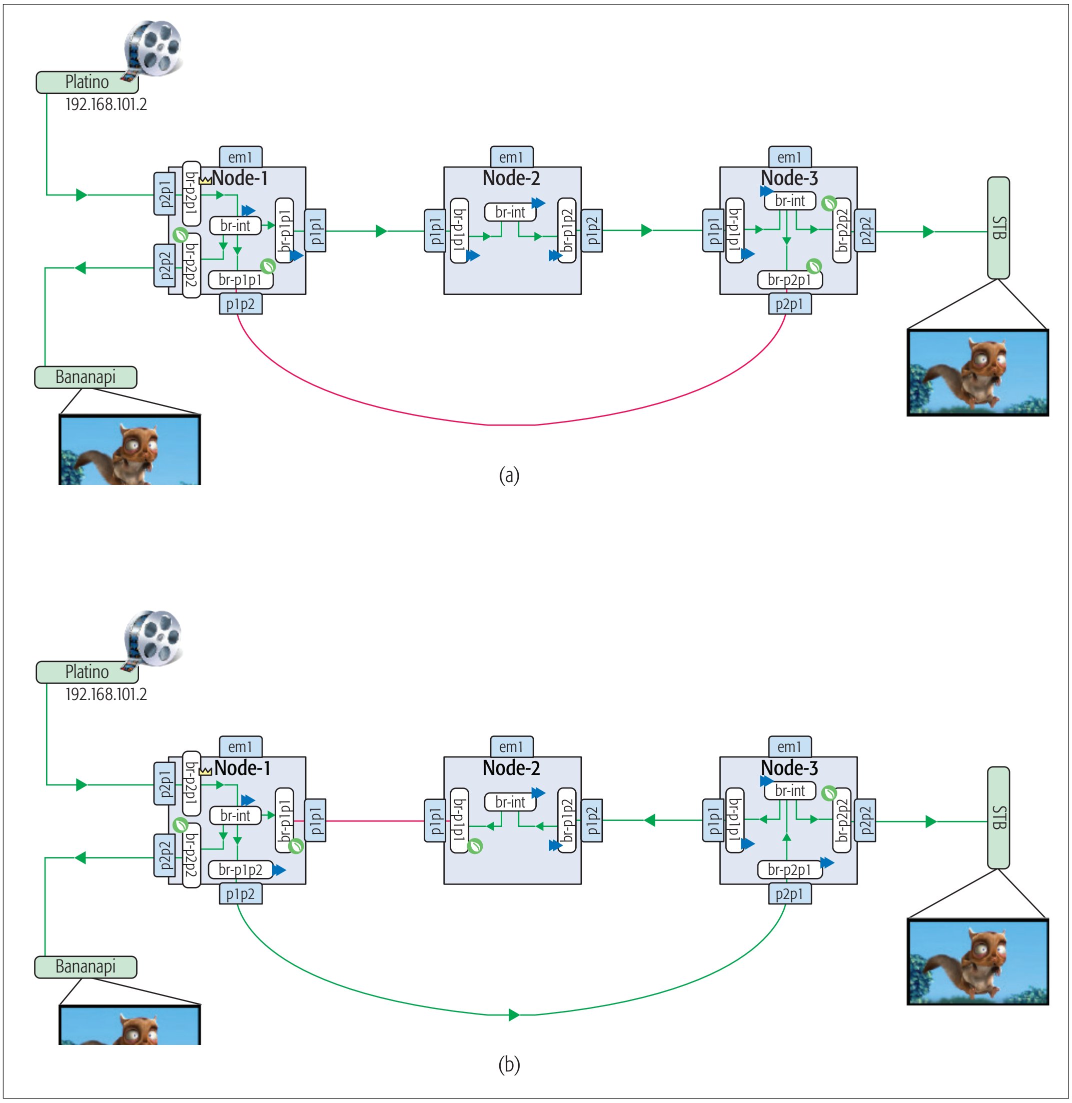

Figure 5. Two distribution trees for different link costs.

previously recorded in an MPEG-2 TS file, in RTP/UDP to an IP multicast address and another for recording an RTP-encapsulated H.264 stream into a file according to the MPEG-2 TS format. The former is executed in the SRC host and emulates an SHO station transmitting a live video stream. The latter is executed in node-3 to dump the received multicast channel and evaluate its quality compared to the original file.

Figure 5 shows two different distribution trees. The tree in Fig. 5a is obtained by assigning a link cost 2 to the link connecting node-1 and node-3. This link (in red in Fig. 5a) is hence discarded by the Kruskal algorithm to build the minimum spanning tree. The tree in Fig. 5b is obtained by subsequently assigning a higher link cost to the link connecting node-1 and node-2. After this change, the link (in red in Fig. 5b) is discarded, and the minimum spanning tree now includes the link connecting node-1 and node-3.

By running the dump script at node-3, we evaluated the effect on the quality of the stream received at node-3 as a consequence of the short service disruption caused by tree recalculation and flow rules enforcement in network nodes. This disruption caused the loss of a few frames during a time interval of less than a second, as shown by the screenshot of the MSU Quality Measurement Tool shown in Fig. 6.

CONCLUSION

In this article we have presented a system for managing multicast distribution trees in a hybrid IP/SDN network. The approach we have pursued is meant to allow ISP operators to gradually introduce SDN management systems into their legacy IP-based infrastructures. In particular, we believe that this approach is particularly well suited for managing IPTV infrastructures in ISP networks, as it allows the implementation of centralized traffic engineering of multicast traffic. We are currently working toward the implementation of more complex traffic engineering procedures in larger-scale scenarios.

ACKNOWLEDGMENT

This work was partially funded by the European Space Agency, within the framework of project Secure Hybrid In Network caching Environment (SHINE), ESA Contract No. 4000118273/16/NL/CLP.

REFERENCES

[1] "Cisco Visual Networking Index: Forecast and Methodology, 2016–2021," , June 2017; https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

[2] S. Agarwal, M. Kodialam, and T. V. Lakshman, "Traffic Engineering in Software Defined Networks," *2013 Proc. IEEE INFOCOM*, Apr. 2013, pp. 2211–19.

[3] C. Janz *et al.*, "Emerging Transport SDN Architecture and Use Cases," *IEEE Commun. Mag.*, vol. 54, no. 10, Oct. 2016, pp. 116–21.

[4] L. H. Huang *et al.*, "Multicast Traffic Engineering for Software-Defined Networks," *IEEE INFOCOM 2016*, Apr.2016, pp. 1–9.

[5] S. Vissicchio, L. Vanbever, and O. Bonaventure, "Opportunities and Research Challenges of Hybrid Software Defined Networks," *SIGCOMM Comp. Commun. Rev.*, vol. 44, no. 2, Apr. 2014, pp. 70–75.

[6] Y. Guo *et al.*, "Traffic Engineering in SDN/OSPF Hybrid Network," *2014 IEEE 22nd Int'l. Conf. Network Protocols*, Oct. 2014, pp. 563–68.

[7] R. Canonico *et al.*, "The Virtual Puppet Master: Adaptive Streaming on Top of an SDN-Enabled Virtual Infrastructure," *Euro-Par 2015: Parallel Processing Wksps.: Euro-Par 2015 Int'l. Wksps.*, Vienna, Austria, Aug. 24–25, 2015, Revised Selected Papers, Int'l. Publishing, 2015, pp. 825–36.

[8] R. D. Doverspike *et al.*, "Structural Overview of ISP Networks," *Guide to Reliable Internet Services and Applications*, C. R. Kalmanek, S. Misra, and Y. R. Yang, Eds, Springer, 2010, pp. 19–93.

[9] C. Thorpe *et al.*, "Experience of Developing an Openflow SDN Prototype for Managing IPTV Networks," *2015 IFIP/IEEE Int'l. Symp. Integrated Network Management*, May 2015, pp. 966–71.

[10] ITU-T Rec. Y Suppl. 16, "Supplement on Guidelines on Deployment of IP Multicast for IPTV Content Delivery," International Telecommunication Union, Geneva, Switzerland, Feb. 2012.

[11] S. Bhattacharyya, "An Overview of Source-Specific Multicast (SSM)," IETF RFC 3569, July 2003; http://www.ietf.org/rfc/rfc3569.txt.

[12] CableLabs, "IP Multicast Adaptive Bit Rate Architecture Technical Report," Oct. 2016; https://apps.cablelabs.com/specification/ip-multicast-adaptive-bit-rate-architecture-technical-report/.

[13] Streaming Video Alliance, "Streaming Video Alliance Announces the Completion of a Multicast ABR Proof of Concept and Welcomes New Members," July 2017; https://www.streamingvideoalliance.org/2017/07/31/multicast-abr-poc/.

[14] S. Schaller and D. Hood, "Software Defined Networking Architecture Standardization," *Computer Standards and Interfaces*, vol. 54, no. 4, Nov. 2017, pp. 197–202.

[15] ETSI ISG NFV, "ETSI GS NFV 001 V1.1.1: Network Functions Virtualization: Use Cases," Oct. 2013.

BIOGRAPHIES

ROBERTO CANONICO (roberto.canonico@unina.it) is an associate professor in the Department of Electrical Engineering and Information Technology (DIETI) at the University of Napoli. His research interests fall in the area of computer networking and cloud computing, and include software-defined networking, network virtualization, resource management and allocation for cloud computing infrastructures, and design of experimental research infrastructures.

SIMON PIETRO ROMANO (spromano@unina.it) is an associate professor in the DIETI at the University of Napoli. He teaches computer networks, network security, and telematics applications. He is also the co-founder of Meetecho, a startup and university spin-off dealing with scalable video streaming and WebRTC-based unified collaboration. He actively participates in IETF standardization activities, mainly in the applications and real time area.

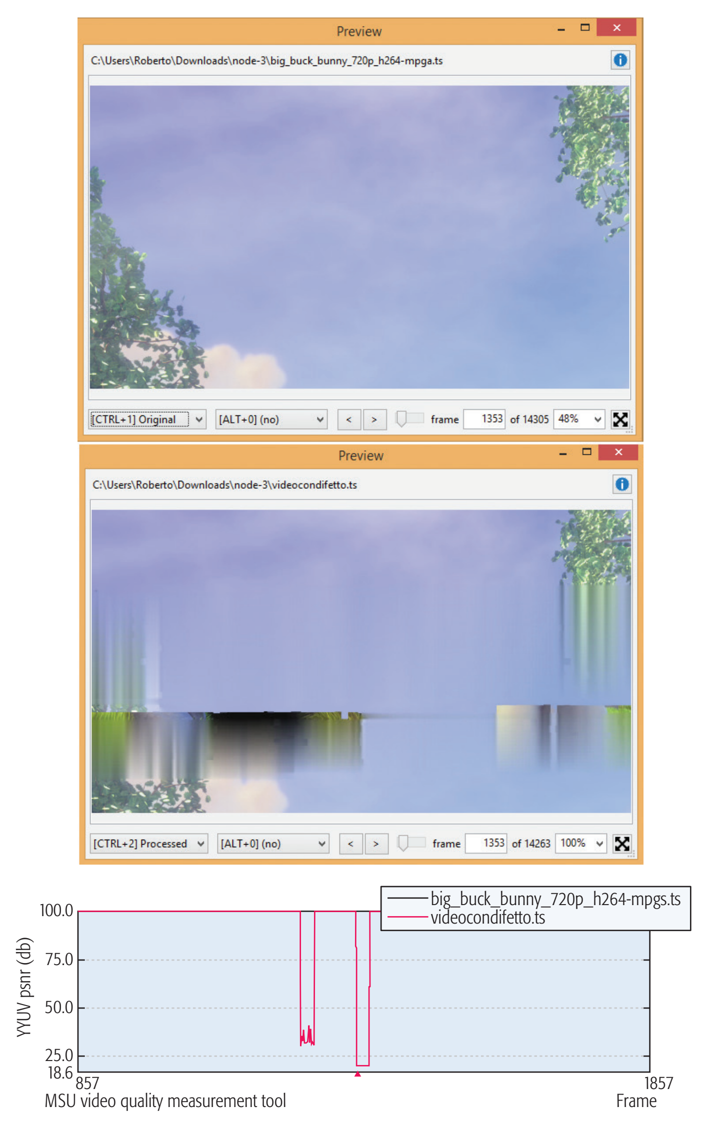

FIGURE 6. Effect on received stream of a tree change.