

A Survey of Multicast in Software-Defined Networking

Weidong Gu^{1,2}, Xinchang Zhang^{2, a}, Bin Gong¹, Lu Wang²

¹Department of Computer Science and Technology, Shandong University, Jinan 250101, China

²Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan, 250101, China

^aemail: zhangxc@sdas.org

Keywords: Multicast; Software-Defined Network; Controller

Abstract. Multicast is an important communication way, which addresses how to distribute the data from one or many sources to a group of destination computers simultaneously. Recently emerging software-defined networking (SDN) separates the network control plane from the data forwarding plane with the promise to dramatically improve network resource utilization. The multicast in this new network paradigm should be reconsidered according to the new network features. In this paper, we introduce the multicast technique in SDN, including SDN-based Multicast in Datacenter, Multicast Routing in SDN and Multicast Tree Packing in SDN. We believe that the multicast technique in SDN, especially the multicast tree packing, has better potential performance than that in traditional network.

Introduction

Multicast is an important communication way, which addresses how to distribute the data from one or many sources to a group of destination computers simultaneously. The typical application examples include video conference, video-on-demand and file distribution. Multicast can be classified into two main types, i.e. IP multicast and application layer multicast (ALM). IP multicast is a technique for one-to-many communication over an IP infrastructure in a network. The nodes in the network (switches and routers) take care of replicating the packet to reach multiple receivers such that messages are sent over each link of the network only once. Therefore the forwarding efficiency of IP multicast is very high. However, it has not been widely deployed in current Internet because of some limitations such as dependence on the supports of network infrastructures and rapid resource-consuming of routers. As an alternative of IP multicast, ALM is implemented at the application layer, using only end-systems. Participating peers organize themselves into an overlay topology, where each edge in this topology corresponds to a unicast path between two end-systems or peers.

Recently emerging software-defined networking separates the network control plane from the data forwarding plane with the promise to dramatically improve network resource utilization, simplify network management, reduce operating cost, and promote innovation and evolution. In SDN, the controller can collect information from network devices and change the traffic flow settings. With the full knowledge of network condition, the SDN controller can adaptively set up different routes for different flows to maximize the service utility. In this paper, we present a survey of multicast in software-defined network.

Typical SDN Architecture

In this section, we review two well-known SDN architectures, i.e. ONF OpenFlow-based SDN and IETF ForCES. The above two architectures each follow the basic principle of separation between the control and data planes, and each standardize information exchange between planes. However, there exist some differences on the architecture design.

The ONF SDN architecture [1] comprises three layers, i.e. data plane, control plane and application plane, as Figure 1 shows. The network elements in the data plane expose their capabilities toward the control layer. In the Controller Plane, the SDN controller translates the

applications' requirements and exerts more granular control over the network elements. Services are offered to applications via the application-controller plane interface. An SDN controller may orchestrate competing application demands for limited network resources. SDN applications reside in the Application Plane, and communicate their network requirements toward the Controller Plane.

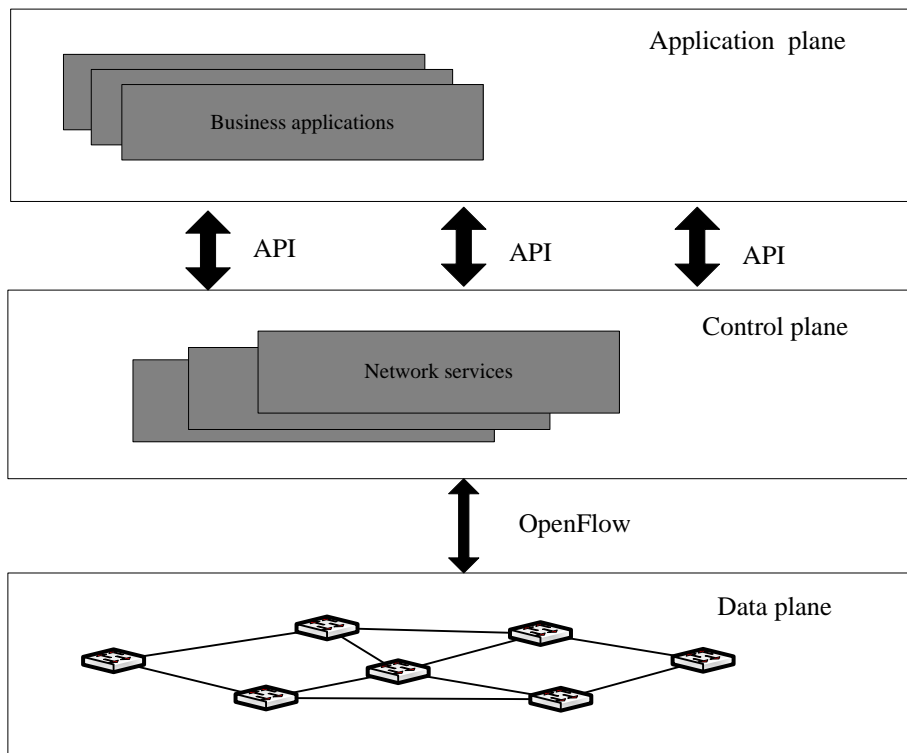


Fig.1. ONF SDN (Source: <https://www.opennetworking.org/>)

The IETF ForCES (Forwarding and Control Element Separation) [2] redefines the network device's internal architecture, in which the control element separates from the forwarding element, as Figure 2 explains. However, the network device is still represented as a single entity. Unlike OpenFlow-based SDN, the control and data planes are kept within close proximity (e.g., same box or room). ForCES defines two logic entities called the Forwarding Element (FE) and the Control Element (CE), which each implement the communication protocol. The FE uses the underlying hardware to provide per-packet handling. The CE executes control and signaling functions and uses the ForCES protocol to instruct FEs on how to handle packets.

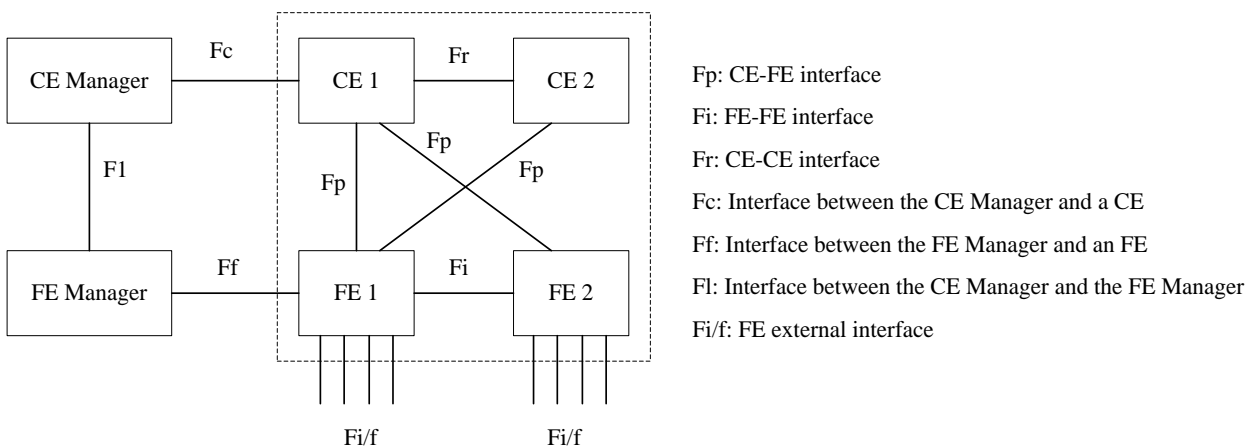


Fig.2. ForCES (Source: RFC 3746)

Multicast in Datacenter

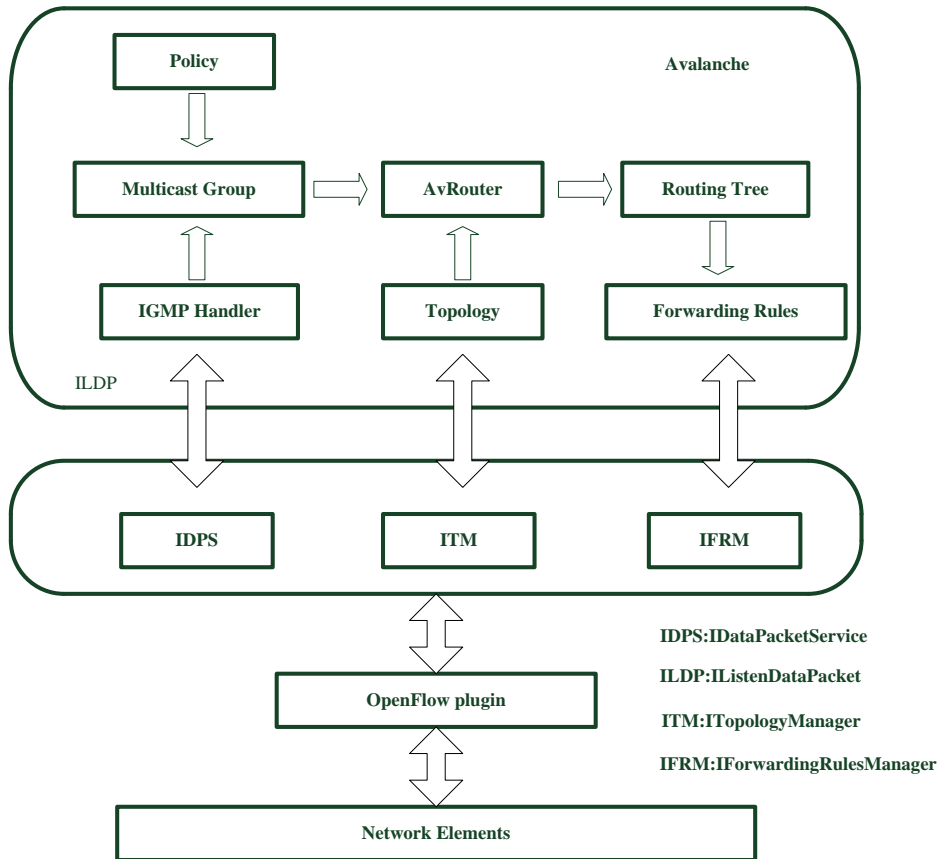


Fig.3. Architecture of the Avalanche OpenDaylight implement

Iyer et al. presented an SDN based system, called Avalanche, which enables multicast in commodity switches used in data centers [3]. Avalanche adopts a new multicast routing algorithm called Avalanche Routing Algorithm (AvRA) that attempts to minimize the size of the routing tree. In typical data center topologies like Tree and FatTree, AvRA tends to solve the Steiner Tree problem. This solution uses the SDN technique to take advantage of the rich path diversity commonly available in data centers networks, and thereby achieves high bandwidth utilization. Fig. 3 presents the architecture of the implementation of Avalanche based on OpenDaylight [4].

In [5], the multicast addressing and routing scale to much larger numbers of multicast groups than in previous designs, while provides greater robustness to switch and link failures. This literature presents a general method for scaling out the number of supported multicast groups. Rather than treating each switch as an independent entity, it leverage ideas from scale-out storage systems to partition the multicast address space and distribute address partitions across cooperating switches. This literature also introduces a novel indirection and rewriting mechanism that aggregates local groups into virtual meta-groups for addressing and routing, and uses local multicast address aggregation to increase the network's group capacity. In addition, this literature provides some mechanisms based on a fast failover through local multicast rerouting, which are resilient and adapt quickly to switch and link failures. The authors implement the above methods using OpenFlow-compliant switches, which support prefix forwarding, multicast addresses, packet rewriting, and a remotely configurable forwarding plane.

Multicast Routing in SDN

The scalability problem in SDN is more serious than that in traditional network because the network traffic is more difficult to be aggregated. To attempt to address the above problem, a new multicast tree for SDN, named Branch-aware Steiner Tree (BST), is proposed in [6]. The BST problem is NP-Hard. [6] presents an approximation algorithm, called Branch Aware Edge Reduction Algorithm (BAERA). BAERA includes two phases, Edge Optimization Phase and Branch Optimization Phase, to effectively minimize the number of edges and branch nodes. In the first phase, BAERA iteratively chooses and adds a terminal node in K to the solution tree $T(V_T, E_T)$ for constructing a basic BST, where V_T and E_T denote the nodes and edges currently in T , respectively at each iteration. Branch Optimization Phase re-routes the tree T to reduce the number of branch nodes. Branch Optimization Phase includes two steps: 1) Deletion Step and 2) Alternation Step. Deletion Step first tries to remove some branch nodes in T obtained from Edge Optimization Phase, and then Alternation Step tries to iteratively move each of remaining branch nodes to its neighbor node.

Jiang et al. employed an Extended Dijkstra's Algorithm to implement the load balance and multicast in SDN [7]. The extended Dijkstra's algorithm considers not only the edge weights but also the node weights for finding shortest paths from a source node to all other nodes in a given graph. Jiang et al. adopt the concept of virtual IP (VIP) for achieving load-balancing. The client just sends a request to the VIP, and the request will be deflected to one of the multiple servers. The basic idea of proposed load balance algorithm is to forward each request to the nearest server with the link load lower than a pre-specified threshold. If all the servers have link loads larger than the threshold, the algorithm chooses the nearest server, which can prevent congestion on the servers. The proposed multicast algorithm is based on the multicast tree construction algorithm using the extended Dijkstra's algorithm for a multicast group publisher to send data packets to all members in the corresponding multicast group.

Multicast Tree Packing in SDN

The network might carry many concurrent multicast sessions. Considering each multicast session in isolation may cause congestion on some links and reduce network utilization. The optimized sharing of network resources (i.e. nodes and links) among multiple coexisting multicast trees is formulated as a packing problem in which the network tries to accommodate all the multicast groups by optimizing the utilization of resources [8]. Most existing multicast tree packing solutions (e.g. [8],[9][10]) attempt to minimize the total link cost based the least cost tree (Steiner tree) whereby they can use limited reserved bandwidth to accommodate as many as possible coexisting multicast sessions. Packing multicast trees fully using available network resource can provide effective user-oriented optimization. However, it is difficult to monitor links' practical traffic and make a global adjustment on the running routing scheme to accommodate new group members and multicast groups.

The SDN technique provides new power for the multicast tree packing because it can monitor links' practical traffic and make a global adjustment on the running routing scheme to accommodate new group members and multicast groups. Zhao et al. proposed an architecture for multi-party video conferencing by utilizing SDN-enabled multicasting [11]. The authors define a SDN-enabled multicast tree construction and packing problem, which find (1) the video rate r_k for each source k , and (2) route for the source-based multicast tree T_k of each source k , such that the system wide network utility $\sum_{k \in K} w_k \log r_k$ can be maximized while the end-to-end delay for each source-destination pair is maintained under a given threshold and the aggregate flow rate on each link is lower than another given threshold. Note that w_k reflects the importance of the receivers obtaining video from source k .

Conclusion

In this paper, we review the two main architecture of SDN, and introduce the multicast technique in SDN, including SDN-based Multicast in Datacenter, Multicast Routing in SDN and Multicast Tree Packing in SDN. In SDN, the controller can monitor links' practical traffic and make a global adjustment on the running routing scheme according to the practical requirement and network conditions. Therefore we believe that the multicast technique in SDN, especially the multicast tree packing, has better potential performance than that in traditional network.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant No. 61272433 and 61472230, and the Shandong provincial Natural Science Foundation of China under Grant No. ZR2013FM010 and ZR2014YL042.

References

- [1] Open Networking Foundation. www.opennetworking.org.
- [2] L. Yang, R. Dantu, T. Anderson, R. Gopal. Forwarding and Control Element Separation (ForCES) Framework. RFC 3746.
- [3] A. Iyer, P. Kumar, V.Mann. Avalanche: Data Center Multicast using Software Defined Networking. 2014 Sixth International Conference on Communication Systems and Networks [C]. Bangalore: IEEE, 2014.1-8.
- [4] OpenDayLight. <http://www.opendaylight.org/>.
- [5] Xiaozhou Li, Michael J. Freedman. Scaling IP Multicast on Datacenter Topologies.Proceedings of the 2013 ACM International Conference on Emerging Networking Experiments and Technologies[C]. New York: ACM, 2013.61-72.
- [6] Liang-Hao Huang, Hui-Ju Hung, Chih-Chung Lin, and De-Nian Yang. Scalable Steiner Tree for Multicast Communications in Software-Defined Networking.arXiv:1404.3454 [cs.NI]
- [7] Jiang Jehn-Ruey, Yahya Widhi, Ananta Mahardeka Tri. Load Balancing and Multicasting Using the Extended Dijkstra's Algorithm in Software Defined Networking. International Computer Symposium, ICS 2014 [C]. Berlin: Springer, 2014. 2123-2132.
- [8] S.Chen, O.Gunluk, B.Yener. The Multicast Packing Problem [J]. IEEE/ACM Transactions on Networking, 2000 8(3) 311-318.
- [9] Jangha Kang, Kyungchul Park, Sungsoo Park, Optimal Multicast Route Packing [J]. European Journal of Operational Research, 2009 196(1) 351-359.
- [10] C.P. Low, N. Wang. An Effective Algorithm for Group Multicast Routing with Bandwidth Reservation. Proc.of the 2nd International Workshop on Autonomous Decentralized System [C]. 2002. 284-293.
- [11] M. Zhao, B. Jia, M. Wu, H. Yu, Y. Xu. Software Defined Network-Enabled Multicast for Multi-Party Video Conferencing Systems. 2014 IEEE International Conference on Communications [C]. Sydney: IEEE, 2014. 1729-1735.