

Customized Service Path Oriented Segment Routing Based Flow Transmission System

Jian-Yang Huang, Peng-Hao Sun, Zhao Dan

National Digital Switching System Engineering & Technological R&D Center
Zhengzhou, China

Email: 15136143225@163.com

Abstract—Current network routing system could not satisfy the increasing traffic diversity and growing demand of customized service path. Based on the service path customizing oriented SR technology and software-based SDN architecture, a prototype of Customized Service Path Oriented Segment Routing based Flow Transmission System is developed, which is designed on programmable routing control platform ONOS. Compared to other routing system, this system could provide service customization support, and also has a better performance on data transmission and controller cost.

Keyword—Software-defined Networking; segment routing; service path customization; control plane; data plane.

I INTRODUCTION

With the rapid development of Internet technology and the emergence of a large number of new applications, more and more services on the network have diversified and personalized service customization requirements. However, IP network routing service capability is limited and determined, which leads to the fact that Internet can only provide a few of the fixed services and obviously cannot meet the growing demand for diversified services. From the point of view of network routing structure and operation mechanism, network topology architecture, model and protocol are redesigned according to the physical rules of "structure decide performance", which directly enhances the network based addressing and routing by facing many high-level network requirements. Enhancement of basic addressing and routing ability will become a key entry point and a breakthrough to such problems.

In existing researches, [2] designed a programmable virtual router framework called "SwitchBlade" based on FPGA. This framework uses a pipelined design that supports pluggable modular design to support user-defined routing functions. However, limited by FPGA resource constraints, the system can only support up to four homogeneous hardware virtual data plane, which leads to a poor scalability. At the same time, Cisco has proposed the Location and Label Separation Protocol (LISP) [3] in the Internet Research Task Force Routing Task Force (IRTF RRG) to solve the problem of routing scalability and to maximize the evolution of the network. To achieve fast and flexible routing customization, [4] proposed the use of GPU powerful parallel computing capability to achieve the efficient IP data packet search and forwarding. However, the scheme uses a dedicated GPU to implement the forwarding engine of the data plane core. Its programmability is limited and it is difficult to meet the

forwarding requirements of non-IP protocols in future network research. In addition, [5] proposed a high-performance software router RouteBricks by cluster technology, but the data path of the scheme is completely constructed on software, so the modification is difficult and the processing speed could hardly meet the requirements.

Based the problems stated above, a Customized Service Path Oriented Segment Routing based Flow Transmission system (CPSRFT) is proposed. The system is based on Software-defined networking (SDN), which is characterized by a loosely coupled control plane and data plane. It supports centralized service customization and control. The underlying network facilities control the upper control decision of transparency and new technology to achieve good compatibility with the existing operating structure. The structure, function and control process of the system are described in detail, and the necessary evaluation and analysis are carried out to verify the effectiveness of the system.

II SEGMENT ROUTING

In order to make the current IP / MPLS network more service-oriented and efficient, in 2013, IETF proposed Segmentation Routing (SR) technology [6, 7]. SR is the source of the core routing, that is, source nodes select the data forwarding path and save it in the packet header. When data is forwarded, the intermediate nodes only need to forward information according to the path of the packet header to process the data packet. The data forwarding path held at the head of the packet is represented by a sequence of segment identifiers (SIDs), where each SID represents a node or a link in the network. When data is forwarded, the intermediate node only needs to process the packet according to the outer SID of the packet header. The specific operation rules are as follows:

- i. If the outermost SID represent the node, delete such SID and move next SID to the head position;
- ii. If the outermost SID represent other nodes, forward such packet to the corresponding node with the shortest path;
- iii. If the outermost SID represent adjacent path of the present node, send this packet using such path.

In practice, SR can be compatible with the existing MPLS implementation, and almost do not need to make major changes to its architecture. In the present study, the MPLS label stack is usually used to store the SIDs of SR.

SR can flexibly determine the forwarding path of data at the source node according to the service requirements. It is a routing technology with strong service path customization capability. Although the traditional MPLS technology also has a certain service customization capabilities, but SR could perform better, mainly in such aspects:

- iv. SR doesn't need signaling protocol to construct forwarding table for intermediate nodes;
- v. SR only saves the status of single flow, which saves the labor such as label distribution protocol (LDP) and Constraint-Based LDP (CR-LDP) in MPLS;
- vi. SR has a light control plane, which need no massive maintenance of layered GMPLS label-switching paths.

In current implementations, the SID of an SR node is often set to the loopback address prefix of the routing node and distributed through the extended Interior Gateway Protocol (IGP). After distribution, an SR forwarding rule table is installed at each node, as shown in Figure 1. The first column of the forwarding rule table indicates the SID number at the top of the packet header label, and the second column indicates the corresponding processing action. When data is transferred, the source node will calculate the data forwarding path into SIDs and the header into the stack, after the intermediate nodes only according to the current top of the stack SID to deal with the data packet. Assume that node A needs to communicate with G along path (A-C-D-F-G) in Figure 1, it embeds the SID sequence (103-106-107) into the MPLS label stack of the header and the current stack top SID is 103. The processing of data packets by intermediate nodes along the way is shown in Table 1:

Node A		Node C		Node E	
Label	Action	Label	Action	Label	Action
102	Pop, Port 2	101	Pop, Port 3	101	Pop, Port 3
103	Pop, Port 1	102	Port 2, Port 3	102	Port 2, Port 3
104	Port 1, Port 2	104	Pop, Port 2	104	Pop, Port 2
105	Port 1	105	Pop, Port 1	105	Pop, Port 1
106	Port 1, Port 2	106	Port 1	106	Port 2
107	Port 1, Port 2	107	Port 1	107	Port 1
1001	Pop, Port 1	3001	Port 2		

Node B		Node D		Node F	
Label	Action	Label	Action	Label	Action
101	Pop, Port 1	101	Port 1, Port 2	101	Port 2
103	Port 1, Port 2	102	Pop, Port 1	102	Port 2
104	Pop, Port 2	103	Pop, Port 2	103	Port 2
105	Port 1, Port 2	105	Port 2	104	Pop, Port 2
106	Port 2	106	Pop, Port 3	105	Port 1
107	Port 1, Port 2	107	Port 3	107	Pop, Port 1

Node G	
Label	Action
101	Port 1, Port 2
102	Port 2
103	Port 1
104	Port 2
105	Pop, Port 1
107	Pop, Port 2

Figure I. SR forwarding rule table

TABLE I. PROCESSING OF SR CONTROLS DATA PACKETS FORWARDING

Node	Current Stack Top	Operation
A	\emptyset	Push {103,106,107}
	103	Pop 103 ,Port 1
C	106	Port 2
D	106	Pop 106 ,Port 3
F	107	Pop 107 ,Port 1
G	\emptyset	Accept

SDN is an emerging software-based network architecture and technology, which has a flexible software programming capabilities, making the network's automated management and control capabilities to an unprecedented increase, which can solve the problem of expansion of limited resources, poor network flexibility, difficulty to quickly meet business needs and other issues that facing current network system. Using SDN centralized control mode to complete the SR data forwarding path calculation and SIDs coding, SR can solve the routing problem and to maximize the consideration of the SIDs coding length [9,10]. Figure 2 shows an SR reference model under the SDN architecture. SR controller can support the global information of network resources and the global deployment and optimization of resources according to the service requirements of the source nodes, such as traffic engineering, load balancing, etc., which achieves a personalized service customization while enhancing the overall network resource utilization. Each node of the data plane is configured with an MPLS packet forwarding interface supporting SR technology. The source node is responsible for interacting with the SR controller to obtain the SIDs of the packet forwarding path. The intermediate node is only responsible for fast data forwarding.

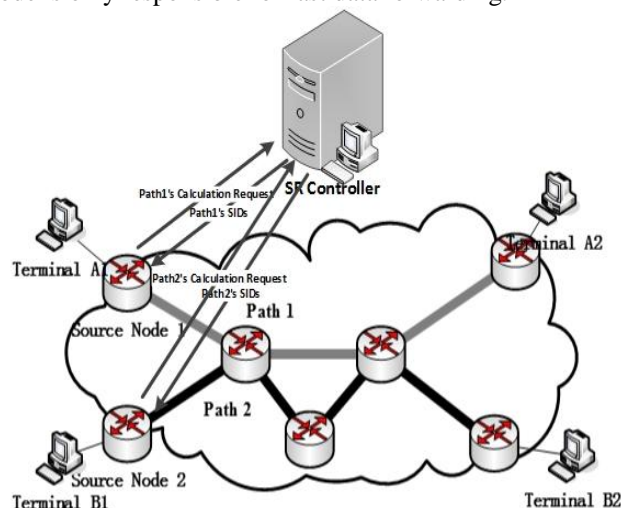


Figure II. SR reference model under the SDN architecture

III OVERALL ARCHITECTURE

SR under SDN architecture can achieve centralized control of network state and support flexible software programming, which has brought superiority of supporting diversified service path customization. In this paper, the following scheme of CPSRFT system is proposed based on SDN architecture, as shown in Figure 3.

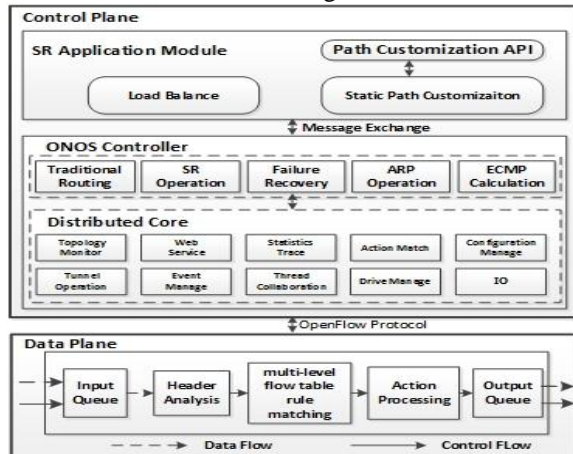


Figure III. CPSRFT system overall architecture

The overall system architecture in Figure 3 can be divided into control plane and data plane. Between them, the control plane mainly includes SR application and ONOS controller. SR application module is mainly responsible for

the realization and maintenance of SR functions, providing service path customization performance guarantee for different business requirements. ONOS controller module is mainly responsible for providing high scalability, high reliability and high stability of resource management and basic function services, which provides a safe and highly available operating environment for SR application modules.

Data plane is mainly based on the rules generated by the control plane configuration for high-speed packet forwarding and processing. It mainly comprises of a header analysis module, a multi-level flow table rule matching module and an action processing module. Header analysis module is used to identify data packet, which extracts meta-data (Src port, MAC src, MAC dst, VLAN id, MPLS, IP src, IP dst, etc.) for match table to compromise match fields. The multi-level flow table rule matching module adopts TCAM + SRAM architecture to search the meta-data extracted by the matching module of the header. The action processing module performs the forwarding, discarding, modification and so on according to the action instruction, whose specific process type is determined by identifying the action flag in the action instruction. Specific actions include forward / output, drop, modify, add / strip VLAN tag, as shown in Table 2.

The action instruction of every bit in Action_flag is shown in Table 3.

Table II. Specific actions

Action Type	Operations
forward/output	Set destination port of additional header according to the field of forward bitmask in actions
drop	To determine whether the field of forward bitmask is empty ,if it is empty ,drop it
modify	Parse packet header and confirm the position of each domain, then modify the packet header according to parameters
strip VLAN tag	Don't pass down after receiving strip VLAN tag
add VLAN tag	Add VLAN tag in front of the packet header

Table III. Action instruction of every bit in Action_flag

- 0: Port forwarding
- 1: Set VLAN ID
- 2: Modify MAC source address
- 3: Modify MAC destination address
- 4: Modify IP source address
- 5: Modify IP destination address
- 6: Modify IP tos domain
- 7: Modify TCP source port
- 8: Modify TCP destination port
- 9~15: Reserved field

A. Control Process of Routing

The routing control process is realized by the control plane of the system, which mainly includes two parts: main communication body announcement and route notification, and service path building.

1) Main communication body announcement and route notification

SR identifies the nodes and links in the network using the node segment SID number and the adjacent segment SID number, respectively, but they are generated in a different way and announced in different ranges.

The node segment SID number is globally valid and can be accepted and processed by all the routing nodes in the entire SR domain. When a new host or routing node joins the network, it first requests the node segment SID number from the SR database (SRDB). Because SDN controllers have centralized network state control capabilities, SRDBs

are typically deployed on the controller. The same node can be assigned multiple SIDs, but the same SID number can only be assigned to one network node. When a node obtains its own SID, it advertises it to the whole network through the IGP extended protocol.

The adjacency segment SID number is locally valid and is used to identify a link in the network. Adjacency segments are generated and assigned SID numbers by the neighboring nodes. The SIDs must be allocated in a way that they cannot occupy the space of the node segment SID number. A node may be assigned multiple SIDs, and the same SID number may be assigned to multiple links, following the same link when allocating adjacent segment SID numbers. Adjacency Segment SIDs cannot be distributed by the IGP extension protocol, but are originated by the originating node and advertised to all neighbor nodes.

After completing the node segment and adjacent segment SID number generation and notification, each routing node will be installed with the forwarding rule table as shown in Figure 1.

2) Service path building

CPSRFT system supports two SR services: load balancing and static path customization. Between them, the static path customization service provides the path customization API to the user, simultaneously calls the SR processing module which the ONOS controller provides, completes the user the path customization request, and achieves the QoS guarantee to the request service. The process is shown in Figure 4.

In Figure 4, the SR manager is the core processing unit of the whole SR application. It receives the input service path information and completes the generation of the data forwarding path SIDs by calling ARP processing, ICMP processing, traditional IP processing and path calculation.

The tunnel management module receives the SIDs information generated by the SR manager and completes the path deployment by calling multiple functional units of the ONOS distributed core.

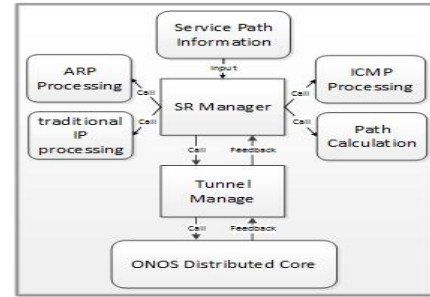


Figure. IV. Processing of static path customization

CPSRFT load balancing service can be considered as a higher level of static path customization service. It calls the path customization API provided by the static path customization service to complete the path deployment. The CPSEFT system needs to allocate traffic on different links according to the current network resource utilization to improve the overall network throughput and reduce the time to complete the flow. Therefore, this paper designed a CPSRFT load balancing algorithm, as shown in Algorithm 1.

When a flow arrives at the controller, algorithm 1 first calculates all its equivalent shortest paths and the available bandwidth of those paths, and then takes the path that has the smallest available bandwidth and also satisfies the requirement as the transmission path of the flow. When there are more than one such path, the path that carries the least flows is selected as the transmission path of the stream.

Algorithm 1 CPSRFT Load Balancing Algorithm

```

Input:  $f$  /*Incoming flow information */
Output:  $path(f)$  /*Transmission paths of  $f$  */

 $P = \text{calculatePath}(f)$  /* Calculate all equivalent shortest paths of  $f$  */
for  $p_i$  in  $P$  do
     $B_{p_i} = \text{MinLinkBandwidth}(p)$  /*Calculate available bandwidth of all paths */
end for

 $P^* = \text{Descsort}(\{ B_{p_i} \})$  /*Sort all paths in descending order according to the available bandwidth */
 $P_{\text{satisfy}} = \text{firstAvaiPath}(P^*)$  /*Obtain the path set that could satisfy the requirement of  $f$  and have less bandwidth */
if  $P_{\text{satisfy}} \neq \text{null}$ 
     $path(f) = \text{MinFlowCount}(P_{\text{satisfy}})$  /*Choose the path of least flow number */
else
    refuse( $f$ ) /*IF all paths are not satisfied, refuse the transmission of  $f$  */
end if
  
```

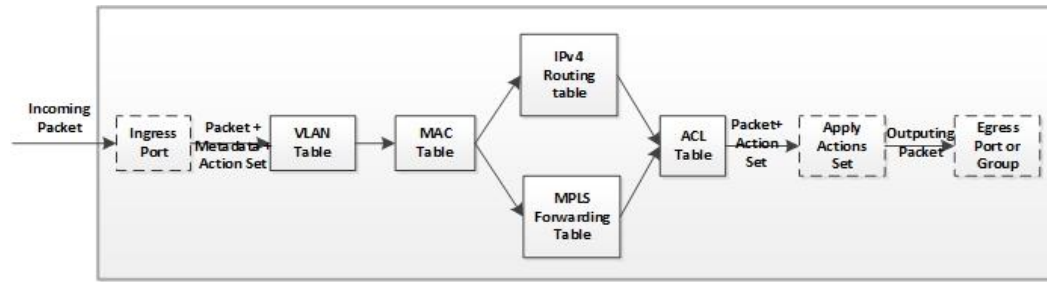


Figure V. SR multi-table pipeline processing structure design

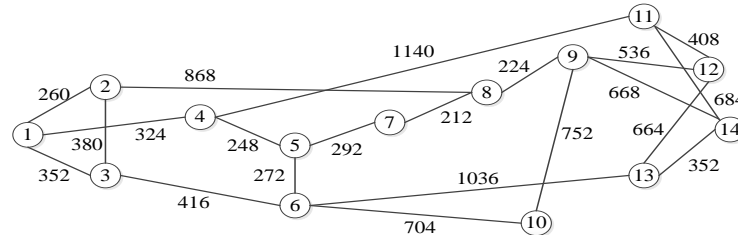


Figure VI. US National Science Foundation backbone network

B. Data Transmission Process

CPSRFT system in the data plane uses OpenFlow multi-table pipeline processing approach to process the data packet. Processing pipeline is a key part in router on the packet forwarding and processing, where different lines carrying different architectures, and also different architectures has different packet processes. In a particular processing pipeline, the data packet is in accordance with the corresponding architecture approach to perform look-up and forwarding. Figure 5 is a multi- table pipeline processing structure design.

When the packet arrives at the incoming port, it will be appended with a set of metadata elements and actions that are used to match in the multi-pipeline table. After that, the packet will start the match operation from the first flow table. During the matching process, the packet can jump from the left flow table to the right flow table, but not from the right flow table to the left flow table. In this system, the first flow table to be matched is the VLAN flow table, which enables the routing nodes in the SR domain to support the VLAN function. The second table to be matched is the MAC table, which will be sent to the corresponding flow table according to the type of the packet. If the packet is a traditional IPv4 packet, it will be sent to IPv4 Routing table; if the data packet with MPLS labels, will be sent to the MPLS forwarding table; if the data packets are not the first two, it will be sent to the controller for further processing. When the packet arrives at the last flow table, all action instructions in its action set will be executed.

IV PERFORMANCE EVALUATION

In order to verify the effectiveness and feasibility of the CPSRFT system, this section evaluates both the efficient transmission of the data and the communication overhead of

the controller. Experimental verification environment using a total of 14 nodes with 21 links of the US National Science Foundation backbone network (NSFNET) as the network topology, as shown in Figure 6. The numbers on each edge in the topology represent the actual physical distance of the neighboring nodes. In simulation, the link bandwidth is set to 100M and the link delay is 10ms. The routing algorithm calculates the optimal path based on the distance between nodes.

A. Experiment of Data Transmission

Based on the experiment environment stated above, we use node 1 in Figure 6 as the service customization client and node 14 as the Web server in the network topology, and then verify the average transmission delay of traditional SDN network and CPSRFT respectively. In the experiment, the request packets of all nodes satisfy the Poisson arrival characteristic, and the average sending rate is 20 packets per second, 1024 bytes for each message, and the simulation time is 1 minute.

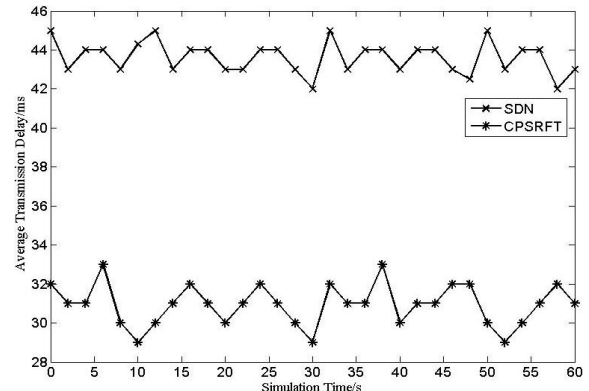


Figure VII. Comparison of average transmission delay

As can be seen from Figure 7, nodes in traditional SDN network do not have the service customization function, the controller does not provide personalized routing customization service for the special node, so the average transmission delay of the network is kept at about 44.2ms. The CPSRFT system has a detailed service path customization capability, which will create a high-speed SR data transmission tunnel for the special service node. Therefore, the data transmission delay of the CPSRFT is significantly reduced and remained at 32.1ms, and the data transmission delay performance enhances by about 27% compared to traditional SDN network.

B. Experiment of Communication Overhead in Controller

In order to evaluate the communication frequency between the control plane and the data plane when the SR control and transmission system is in the process of data transmission, the communication behavior between data plane and control plane of traditional SDN network routing mechanism and CPSRFT is counted.

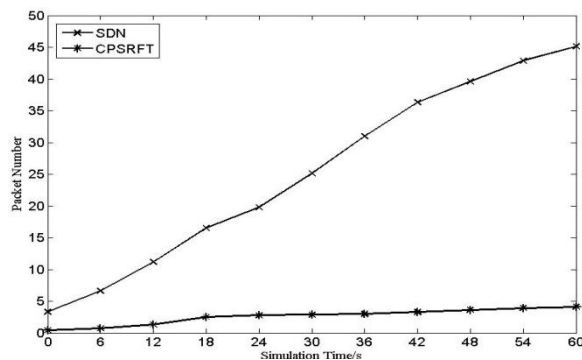


Figure VIII. Comparison of average number of communication

Figure 8 shows the statistics of the total number of control and data plane communication behavior collected with Wireshark. Such number is only the statistics of the additional packets of two different routing mechanisms when supporting data transmission. As can be seen from the figure, since in traditional SDN network, in each data transmission controller should distribute flow table to every corresponding nodes, so the average number of communications in 1min has reached about 43 times. In CPSRFT, the controller only need to send data to the source node SIDs about path transmission sequence, so the total number of communications after 1min is only four, and the controller communication overhead performance compared to the traditional SDN network increased by about 90%.

V CONCLUSION

Current network routing system could not satisfy the increasing traffic diversity and growing demand of customized service path. Based on the service path customizing oriented SR technology and software-based SDN architecture, a prototype of customized service path oriented segment routing based flow transmission system is developed, which is designed on programmable routing control platform ONOS. Compared to current SDN network, this platform could provide service customization and at same time improve data transmission and controller cost performance, which has a great significance on diversified and customized network service in the future.

ACKNOWLEDGEMENT

This work is supported by the research (Grant No. 2013CB329104), and National Natural Science Foundation of China (Grant No. 61521003, 61572519, 61502530, 61672471), and the National High-Tech Research & Development Program of China (Grant No.2015AA016102).

REFERENCE

- [1] Lan J L, Cheng D N, Hu Y X. Research on reconfigurable information communication basal network architecture. *J. Commun*, **35**, (2014).
- [2] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks. *ACM Sigcomm Comput Commun Rev*, **38**, (2008).
- [3] Farinacci D. Locator/ID separation protocol (LISP), <https://tools.ietf.org/id/draft-maino-lisp-sec-00.html>, (2007).
- [4] Han Sangjin, Jang Keon, Park Kyoung Soo, et al. PacketShader: a GPU-accelerated software router. *ACM Sigcomm Comput Commun Rev*, **40**, (2010).
- [5] Dobrescu M, Egi N, Katerina J, et al. RouteBricks: Exploiting parallelism to scale software routers. *ACM Symp SOSP*, (2009).
- [6] C.Filsfils et.al, Segment Routing Architecture, <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-09>, (2016).
- [7] C.Filsfils et.al, Segment Routing with MPLS data plane, <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-mpls-05>, (2016).
- [8] Dotaro E, Vigoureux M, Papadimitriou D. Multi-region networks: generalized multi-protocol label switching (GMPLS) as enabler for vertical integration. *Global Telcommun*, (2004).
- [9] Giorgetti A, Paolucci F, Cugini F, et al. Dynamic restoration with GMPLS and SDN control plane in elastic optical networks. *J. OSA*, **7**, (2015).
- [10] Sgambelluri A, Giorgetti A, Cugini F, et al. OpenFlow-based segment protection in Ethernet networks. *J. OSA*, **5**, (2013).