

## SDN Mixed Mode Flow Table Release Mechanism Based on Network Topology

Liu Chuan<sup>1,2,a</sup>, Hao Hanyong<sup>3,b</sup>, Li Binglin<sup>1,2,c</sup> and Wang Congying<sup>4,d</sup>

<sup>1</sup>Global Energy Interconnection Research Institute, No.8 Nanrui Road, Nanjing, P.R.China

<sup>2</sup>State Grid Laboratory of Electric Power Communication Network Technology, No.8 Nanrui Road, Nanjing, P.R.China

<sup>3</sup>State Grid, No.86 Chang'an Avenue, Beijing, P.R.China

<sup>4</sup>Jiangsu Vocational Institute of Commerce, No.104 Guanghua Road, Nanjing, P.R.China

<sup>a</sup>liuchuan@geiri.sgcc.com.cn, <sup>b</sup>haohanyong@sgcc.com.cn,

<sup>c</sup>libinglin@geiri.sgcc.com.cn, <sup>d</sup>13106857@qq.com

**Keywords:** Software Defined Networking, Flow Table Release Mode, Proactive Mode, Reactive Mode

**Abstract.** The technology of software defined networking, which introduces the concept of centralized control, provides a new idea for the intelligent network. Meanwhile, it also introduces the control delay in the end-to-end transmission delay of the service. Industry network has extremely high requirements on real-time and reliability. Thus, the balance between real-time and flexibility is one of the key points in the application of SDN technology in such communication networks. The article combines the advantages of two types of flow table release mode, and through the method of topology determination, the threshold module is used to realize the flexible and fast switching of the flow chart release mode, which improves the real-time and reliability of the network.

### Introduction

Software Defined Networking as a new type of network architecture was originally proposed by the United States Stanford University Clean State group. It was designed to address the problem of failing to take advantage of large-scale real traffic and rich applications in existing networks, while studying how to improve the network speed, reliability, efficiency and security issues. With the deepening of research [1], SDN has gradually been recognized by academia and industry as the mainstream of the future development of the internet.

### Two Mechanisms of Flow Table Release Mode

In the SDN Network, there are two mechanisms of flow table release mode, which called proactive mode and reactive mode respectively. Proactive mode is issued in the transmission of data before the issue of flow entry. [2] RESTful style Web calls using the HTTP protocol at the far end as well as the SDN controller APP using the embedded API directly under the flow table could be used in the process of flow table release, as is shown in Fig.1.

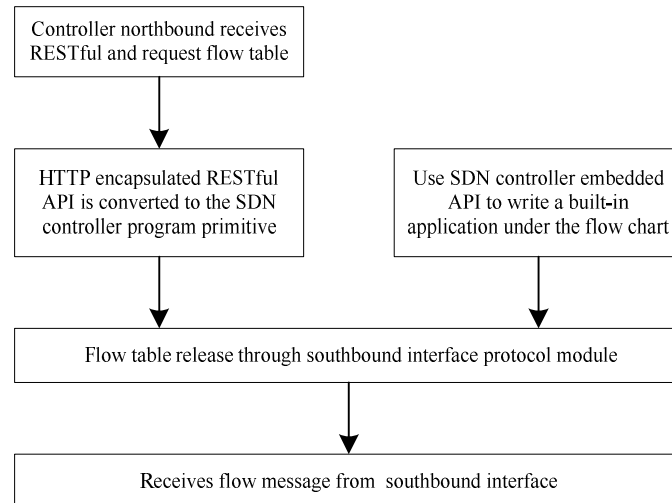


Fig.1 Process of Proactive Mode Flow Table Release

Due to the data transmission has been set before the rules and issued to the corresponding equipment, proactive mode of flow table release is suitable for SDN networks which can be planned in advance. Thus, such mode is more fast. [3] However, in the case of frequent changes in network, proactive mode will result in a huge amount of computation in centralized control system. Reactive mode of flow table release is issued when the switch receives the first packet, for there is no relevant flow entry, the packet will use packet-in message sent to the controller processing, as is shown in Fig.2.

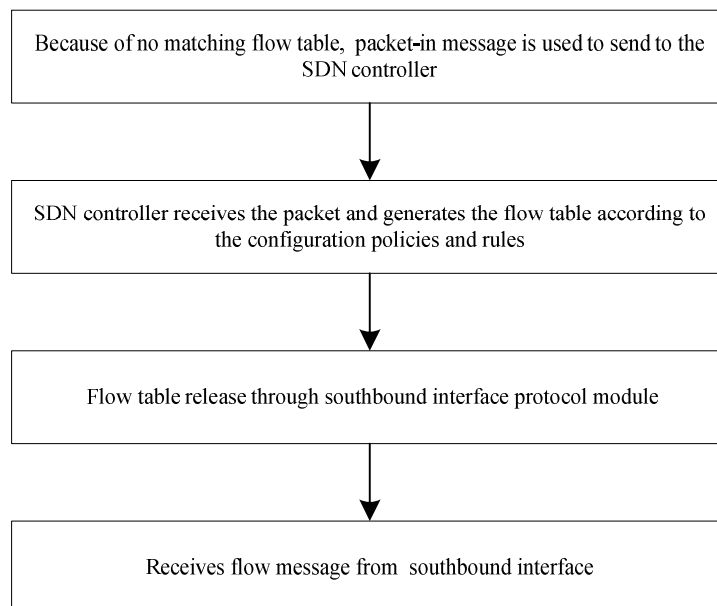


Fig.2 Process of Reactive Mode Flow Table Release

When the controller determines the corresponding processing methods, the relevant flow entry will be issued to the switch and it can be forwarded in next time. The velocity of reactive mode is mainly related to the response time of controller receiving the packet-in message and generating the relevant flow table rules. The mechanism of the reactive mode is defined by the protocol of SDN south bound interface. All of the forwarding rules are completed by the controller.

### Implementation Process of Mixed Mode Flow Table Release Mechanism

If the end-to-end packet is forwarded, all the switches of the packet passing through the path have passed the proactive mode through the controller and the flow table is active. [4] The timing of the

controller flow table will not affect the end-to-end delay of the service. If the packet in the first OpenFlow switch in the flow table match misses, and the Table-Miss rule of the switch is forwarding unmatched packets to the controller, it will trigger the reactive mode, as is shown in Fig.3.

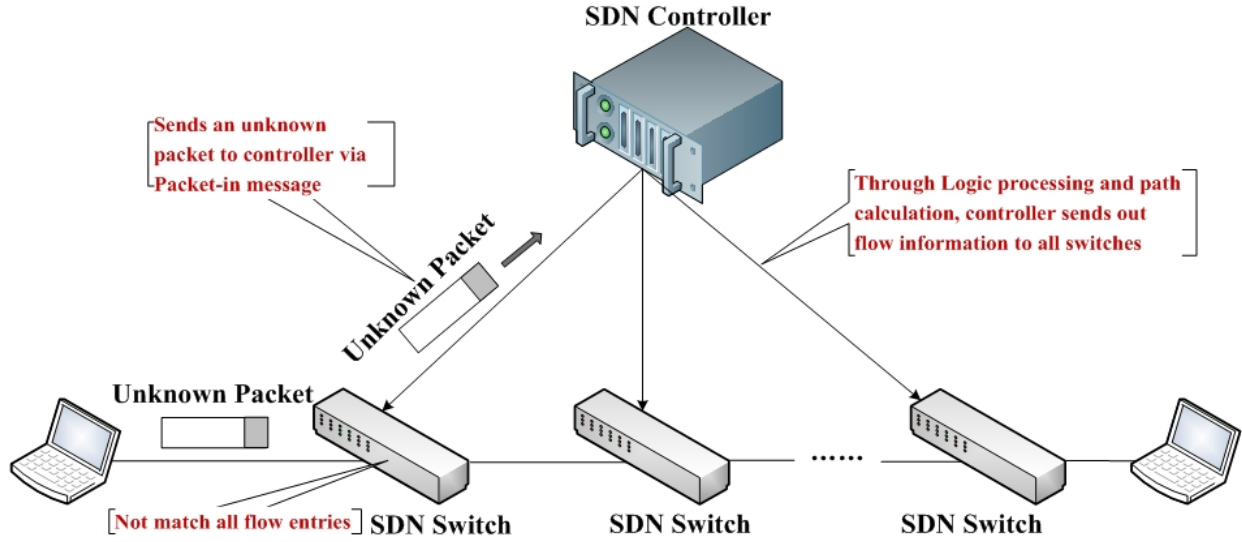


Fig. 3 The Matching Process of Unknown Packet in SDN Network

Combined with the advantages of two types of flow table release mode, SDN Mixed Mode Flow Table Release Mechanism Based on Network Topology is proposed in the article. Its implementation mechanism is divided into the following steps:

A. The application system invokes the SDN controller information through the northbound interface. All the services through the proactive mode flow table release mechanism and update the network topology and flow table rules with cycle  $\frac{T}{n}$ .

B. When the topology changes within period  $\frac{T}{n}$ , the topology refresh and the flow table update period change to  $\frac{T}{n+1}$ . If the topology doesn't change, the period is maintained.

C. Set the topology threshold module in the application system. When the topology refresh period change to  $T_c = \frac{T}{n+k}$  ( $k=1,2,3,\dots$ ), it is determined that the network topology is currently changing state  $S_{T_c}$ , otherwise, it is normal state  $S_T$ .

D. When the topology doesn't change during  $T_c$  cycle, the topology refresh and the flow table update period become to  $\frac{T}{n+k-1}$ , then the process is repeated until the period is  $T$ .

E. When the topology changes during  $T_c$  cycle, the next refresh cycle changes to  $T'_c = \frac{T}{n+k+1}$ . And so on.

F. In the application system for the transmission of the service, according to the importance of marking the priority, and all the aging time of flow table is set to  $\infty$ .

G. When the topology threshold module determines that the network state is  $S_T$ , the service forwarding rule is executed in proactive mode. When the topology threshold module determines that the network state is  $S_{T_c}$ , aging time of the lowest priority service changes to 0, and which of the second lowest priority service changes to  $T_c$ .

H. When the aging time is 0, the schedule of the flow table for the service is invalidated in the application system. In the case, the forwarding rules of the service are executed in reactive mode.

I. When the network topology doesn't change within one cycle  $T_c$  in state  $S_{T_c}$ , the aging time of the flow table of the lowest priority and second lowest priority service revert to  $\infty$ .

J. When the network topology changes within one cycle  $T_c$  in state  $S_{T_c}$ , the topology refresh and the flow table update period become to  $\frac{T}{n+k-1}$  in state  $S_{T_c}$ , as a result of experiencing refresh cycle  $T_c$ , the aging time of the second lowest priority service flow table changes from  $T_c$  to 0, and the service will be converted to reactive mode execution. At this point, the aging time of the third lowest service flow table changes to  $T_c'$  and so on.

For ease of understanding, assume that the current SDN network application system has four kinds of service  $b_1$ ,  $b_2$ ,  $b_3$ ,  $b_4$ , the aging time is  $t_b$ , The implementation process of mixed mode flow table release mechanism is shown in Fig.4.

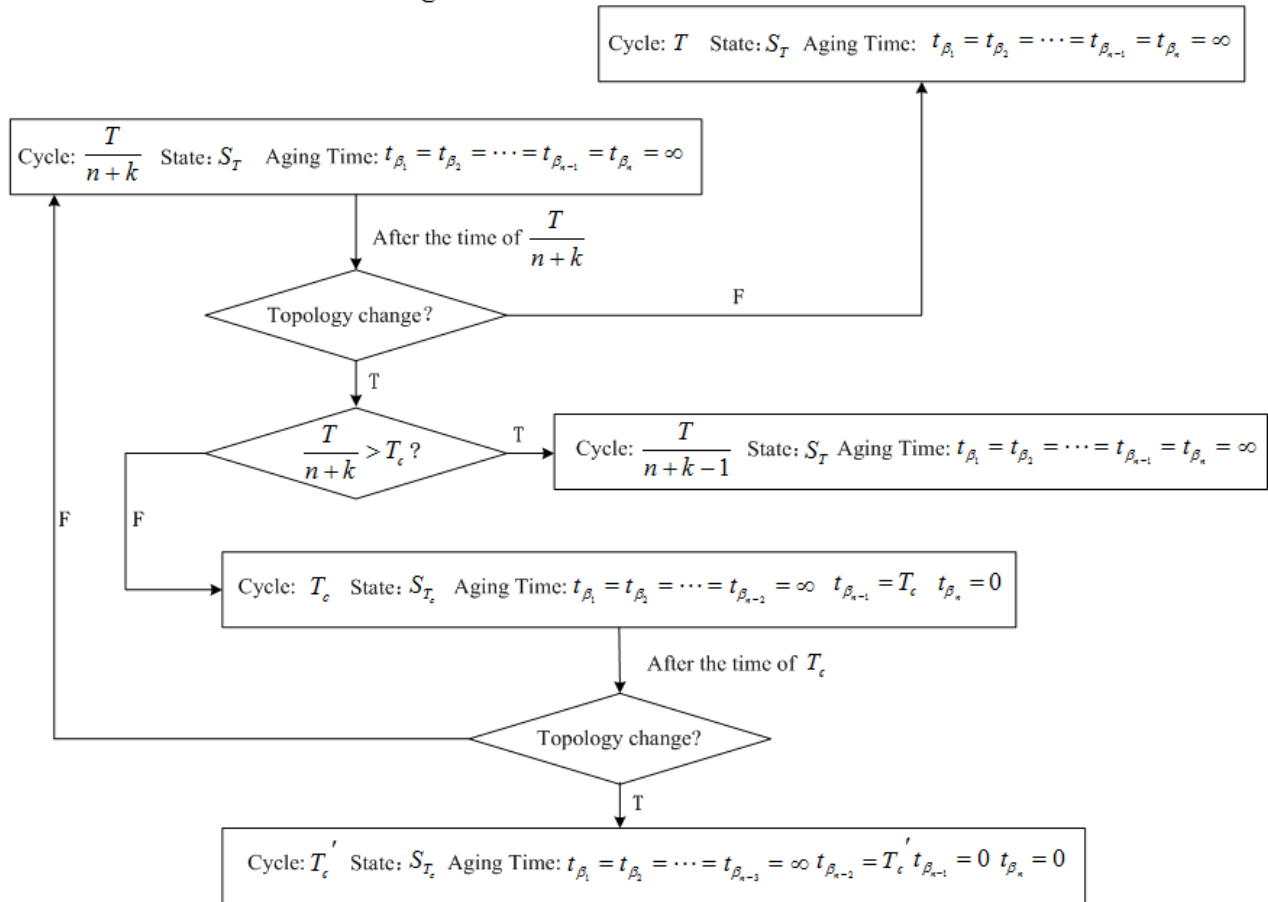


Fig. 4 Process of Mixed Mode Flow Table Release Mechanism

## Simulation Verification

In order to examine the degree of protection of the service using mixed mode flow table release mechanism, the article is experimented with ONOS controllers and Mininet software switch to build simulation environment. Set the initial topology refresh and flow table updating period to 20s and the topology threshold trigger period to 10s. Meanwhile, the priority order of the four services is set to follow  $b_1 > b_2 > b_3 > b_4$ . In the initial state, the four services are set to proactive mode. The simulation result is shown in Fig.5.

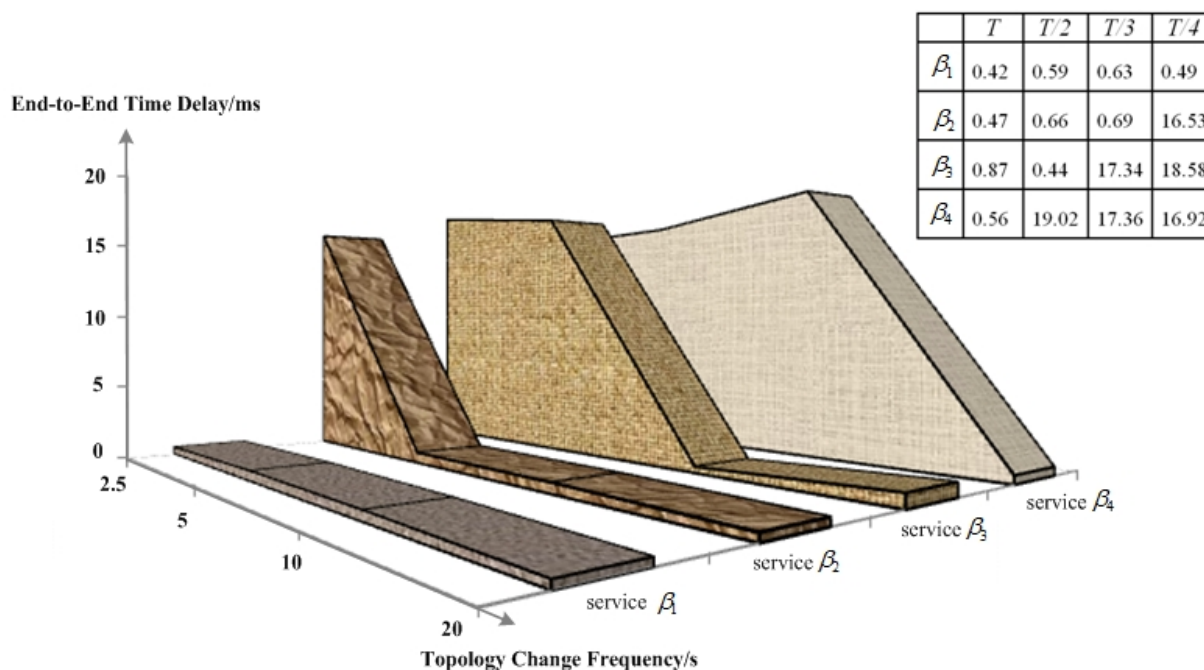


Fig. 5 Simulation Result of Four Services

Through simulation, it can be concluded that when the topology reaches the frequent change state, the high priority service can keep the low latency forwarding of the proactive mode and forwarding strategy of low priority service to the controller automatically. Thus ensuring that high priority service policies can be effectively and quickly implemented when network topology changes frequently.

## Conclusion

In the article, by setting the topology threshold module in the application system, prioritizing the service, invoking the topology information of the controller by northbound interface and transforming real-time transmission mode towards loop decision, mixed mode flow table release mechanism is achieved. It can effectively control the large-scale SDN network due to changes in topology caused by a huge amount of software operations, while ensuring critical service real-time and reliability.

## Acknowledgements

This work was financially supported by Key Technology Research of Software Defined Optical Network Oriented to Multi-domain Interaction of Power Communication Project(5455HT160015).

## References

- [1] Zhang Chaokun, Cui Yong, Tang Heyi, et al. State-of-the-art survey on software defined networking (SDN)[J]. Journal of Software, 2015,26(1):62-81.
- [2] Wang Jiye, Liu Chuan, Wu Junmin, et al. Research of Software Defined Service Resource Equitable Allocation Technology of Power WAN[J]. Power System Technology. 2015, 39(1), 1425-1431.
- [3] D.Kreutz etc., Software-Defined Networking: A Comprehensive Survey, IEEE, Proceedings of the IEEE Volume 103, 2015.1

- [4] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74.
- [5] Liu Chuan, Huang Hui, Yu Qiang, et al. Research of High Reliability Business Support Mechanism for Power Communication System Centralized Control Based on SDN[J]. Electric Power Information and Communication Technology. 2015,13(12),1-5.
- [6] CAO Xiu, HUANG Han, WANG Xueping, et al. Software Defined Grid: Concept, Architecture and Samples[J]. Automation of Electric Power Systems,2016,40(6):1-9.
- [7] ZUO Qingyun, CHEN Ming, ZHAO Guang-song, et al. Research on OpenFlow-Based SDN Technologies[J]. Journal of Software. 2013,24(5),1078-1097.