

CODING OPENFLOW: ENABLE NETWORK CODING IN SDN NETWORKS

Jun Yang, Bin Dai, Lu Lv, Guan Xu

School of Electron. Inf. & Commun, Huazhong University of Science and Technology,
Wuhan, Hubei, China

ABSTRACT

There is no doubt that network coding is a promising enhancement of routing to improve network throughput and provide high reliability. However, there are several open problems in practical network coding, especially on how to guarantee coding advantage for a decentralized control network without the knowledge of the network topology. The biggest benefit of OpenFlow is to decouple the control plane from the data plane, allowing the centralized forwarding decisions in comparison to traditional distributed control network. As a result, we propose a Software-Defined coding network and address key technical challenges in practice. We design NC-OF, a framework to enable networking coding in SDN networks, and use MMF-NC coding strategy proposed by Guan Xu in NC-OF. Finally, we proved that our solutions can effectively improve network performance through simulation experiments. And we also find that network coding is not necessary when the link bandwidth is enough, because it will bring the problems of time delay, the increase in the amount of calculation and so on.

KEYWORDS

Network coding, Open Flow, Coding advantage

1.INTRODUCTION

Since Andswede firstly introduce network coding (NC) into network communication, there is no doubt that network coding is a promising enhancement of routing to improve network throughput and provide high reliability. Currently, NC has been widely studied in areas such as P2P Ad-hoc network, sensor network, sharing system, distributed file storage and network security and so on [1] [2].

However, a very important challenge in network coding research is that studies about realization of NC and real applications are hard to see. Because when applying network coding in practical network, there will be some of the following problems. Firstly, the computation complexity and dynamic topology changes make few coding strategy can be applied to real network. Besides, when coding the packets, it increases transmission delay and extra resource expense of nodes. These disadvantages limited the research of NC mainly in coding theory.

According to the implementation process of NC, it can be divided into Centralized Network Coding and Distributed Network Coding. Distributed NC is used widely because it has good topological adaptability. But relatively it needs more communication overhead.

The Centralized Network Coding can construct coding vector effectively, ensuring the successful decoding of the sink node. It has the advantages of good stability, cheap communication costs. But because it is a centralized algorithm, we need to have a full understanding of the topological structure of the network topology changes. So the application area is limited. Above all, there is no appropriate way to realize the NC.

In this paper, we use OpenFlow model to tackle the above-mentioned problems. OpenFlow model was introduced by Clean Slate Project (Stanford University) [3] with the basic idea that switching nodes only implement the data plan by a set of general flow tables which can be accessed and modified from remote controllers via standard OpenFlow protocol [4].

We propose OpenFlow Network Coding (OFNC), an architecture featuring a clear separation between a data and a control plane, and OpenFlow protocol based on Network Coding (NCOF) the core component of the architecture as a standard communication protocol between the two planes. This architecture is depicted in Figure 1. The controller monitors the Openflow switches in daemon and calculates the coding and routing strategies on receiving routing requests from NC switches, and then control switches to perform coding strategy through the flow table.

Because of the centralized control characteristics of OpenFlow network, it is flexible and easy to manage network routing strategies and switch functions. Packet forwarding will be converted into flow forwarding by OpenFlow, where packet forwarding is controlled completely by the switch and flow forwarding is based on the flow table. In details, routing control is separated from the forwarding facility, and it depends on a centralized controller.

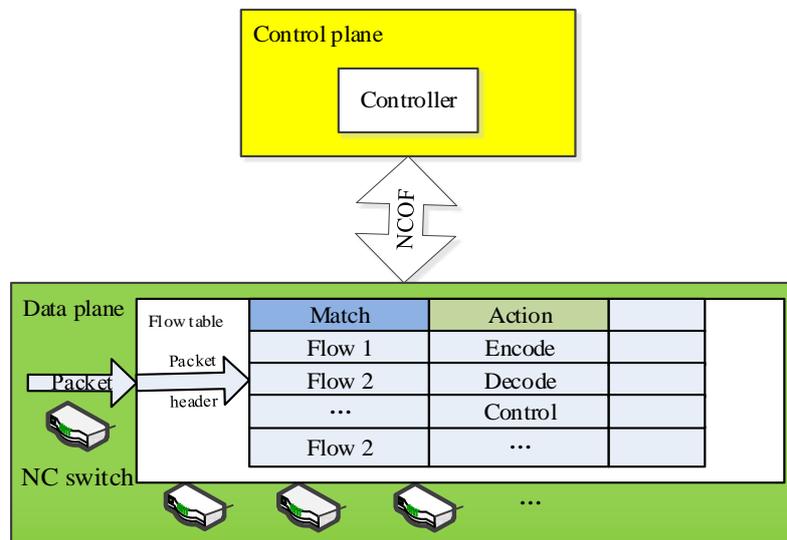


Figure 1. OpenFlow Network Coding Architecture

So the Centralized Network Coding can be realized in this way. In order to ensure that all the changes in the network can be informed in time, the NC switch and OpenFlow controller maintain heart beat communication through NCOF. Once the topology changes, the OpenFlow controller can actively refresh the coding strategy in time, and then set the corresponding switches by modifying flow tables.

From the new design perspective, the problems in implementation of NC can be further solved. In 2012, Felicián Németh et [5] made the first attempt at applying Network Coding in OpenFlow network, demonstrating the feasibility of using network coding in OpenFlow networks.

However, they just extended the OpenFlow switch's experiment module only for butterfly-based NC in the butterfly topology, could not be applied to the actual network. There is still a lack of work on NC routing in OpenFlow controller.

In this paper, We design NC-OF, a framework to enable networking coding in SDN networks, and use MMF-NC coding strategy proposed by Guan Xu in NC-OF. Finally, we proved that our solutions can effectively improve network performance through simulation experiments. And we also find that network coding is not necessary when the link bandwidth is enough, because it will bring the problems of time delay, the increase in the amount of calculation and so on.

Referring to the demonstration proposed in [5], we introduce a software-defined coding system, use MMF-NC coding strategy in network and extend Openflow protocol to include new actions, and add networking coding related functions in controller and switches. In section 2 we describe the technical challenges of the system. Section 3 proposes the solutions. In section4, we propose a framework (called NC-OF) for realizing network coding over SDN network and evaluate the benefits of our framework. Section 5 gives the simulation results of our framework.

2. TECHNICAL CHALLENGES

OpenFlow divides the original system into data plane, control plane and application of plane, providing a programmable network, which revolutionizes the existing network architecture. By this facilitates, NC can be added into OpenFlow network. However, it brings some technical challenges as well.

2.1.CODING STRATEGY

After we add the NC module into OpenFlow switch, how to choose the coding path and the coding packet? This is still a challenge. To solve this problem, we have to focus on how the controller should make the coding strategy, and how to provide NC based routing strategies for the corresponding OF switches.

2.2.EXTRA COST

In the previous paper, it is summarized some problems caused by NC, such as the transmission delay, the information synchronization and extra resource consumption and so forth. When NC is applied into OpenFlow network, two of the following problems should be highlighted.

2.2.1.TRANSMISSION OF CODING INFORMATION

When the OpenFlow switch perform flow coding action depends on the flow table, it reads the packet header, parses the related coding information, on which the receiving packet can be matched through the coding flow table and processed then. However, in OF protocol of the latest version, the packet head field cannot attach such information directly. Thus we need to add this information and do our best to reduce the coding information that must be transmitted.

2.2.2.COMPUTATION COST

There is nontrivial computation cost generated by the operations related to NC such as adding the packet header, looking up the buffered coding packet and packet encoding/decoding. Therefore, in order to guarantee the rate of transmission, we must reduce the computation cost in maximum.

2.3.MEET DIFFERENT QOS REQUIREMENTS

The transmission requirements of different networks are different. The NCOF should meet the requirements such as high real-time and high stability transmission. It needs provide Quality of Service (QOS) routing to satisfy many application demands.

3.PROPOSED SOLUTIONS

In this section, we provide a suite of preliminary solutions to the issues identified above, based on the latest OpenFlow specification [6].

3.1.CODING STRATEGY

The controller get the information of topology and packets from the switches. On the basis of these information, the controller formulates the coding strategy. With the max-min fairness(MMF)as a commonly theoretical support, we use a coding strategy named MMF-NC[7] proposed by Guan Xu, based on Participatory networking[8] in SDN. MMF can solve the fair maximum flow problems in multi-unicast networks by fairly allocating available bandwidth to network flows . However existing solutions only consider available links on paths of existing flows directly, ignoring other links that may provide potential of higher fairness. So G.XU propose a network coding based MMF routing algorithm to address the fair maximum flow problems by considering potential idle links. When searching for network coding routings with the butterfly and grail structure [9] remedy paths are chosen by the shortest path algorithm.

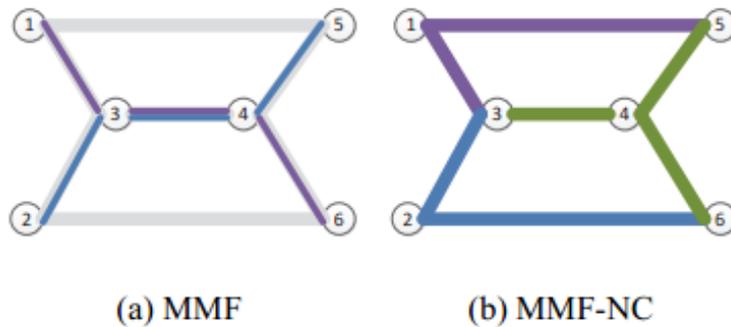


Figure 2 :In a butterfly structured topology of 6 nodes, one flow is generated from node 1 to node 6 in purple, and the other is from node 2to node 5 in blue. The encoded flow is in green.(a)With MMF algorithm, the link(3,4)is the bottleneck;(b)with MMF-NC algorithm, the bottleneck doesn't exist

With the inter-session network coding routings the fairness may be improved. For example, in a directed acyclic network of butterfly structure with link capacity 1in the Fig. 2, two flows with unlimited demands are routed with general MMF and MMF-NC algorithm separately. The fairness for each flow is 0.5 for MMF algorithm, shown in Figure 2(a), with the saturated link(3,4) shared by the two flows and other links " idle" ; while the fairness of each flow will be 1, in the latter algorithm, as the two flows are suitable for inter-session network coding routing, shown in Figure 2(b), replacing two original flows with a encoded flow on the link (3,4), which eliminate the saturated link.

3.2.REDUCE EXTRA COST

In the OpenFlow protocol, it allows to add MPLS label to packet header, so we use this flexible field to transmit corresponding coding information within the coding packets. We send the NC metadata in MPLS labels following the format showed in Figure 3 and Figure 4. The Figure 3 is the encoding packet header and the Figure 3 is the decoding packet header. They all include the flow ID and the corresponding sequence number. In order to reduce the additional transmission, there is only one flow that carries the coding information among all the coding flows. And the coding strategy was transmitted through NCOF, which is similarly with OpenFlow Channel.

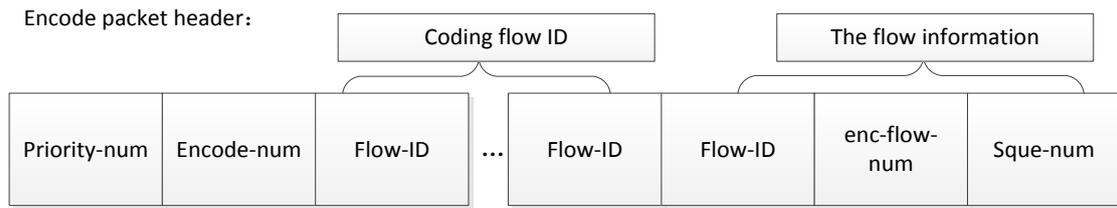


Figure 3. NC encoding header in OpenFlow

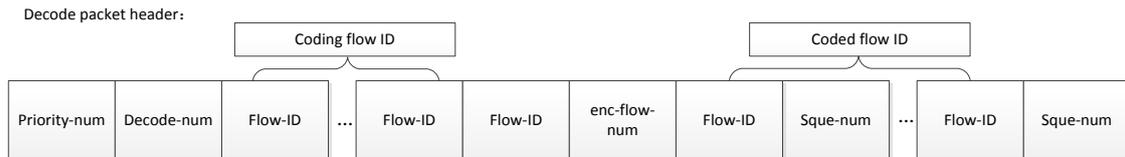


Figure 4. NC decoding header in OpenFlow

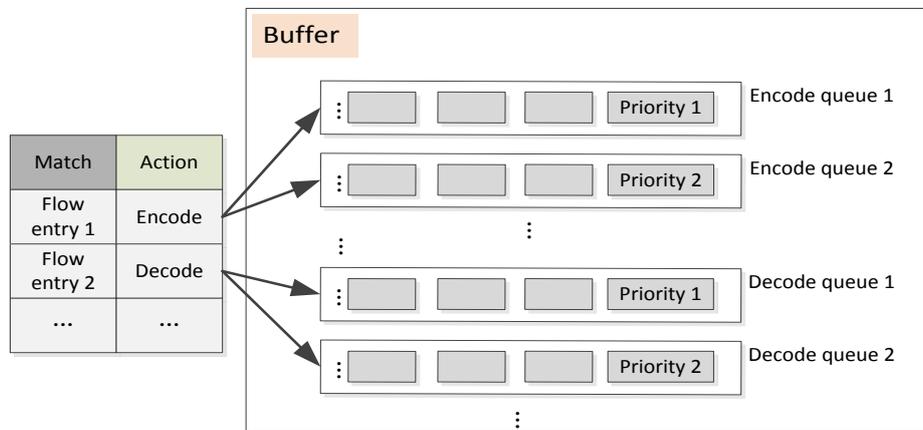


Figure 5. Apply MPLS label to QOS in packets

In addition, we use the XOR coding method to reduce the length of packet header, because this method does not require any coding coefficient. Beside that, due to the XOR operation is simple, it can effectively reduce the time delay caused by coding.

Another method for reducing time delay is optimizing the buffer management. First, we make the openflow switch store the packets respectively by dividing the buffer into encoding queue and decoding queue. Second, the openflow switch extract the coding information brought by the packets from buffer and put the information into linked lists. Then the switch could match the packet through the linked lists directly. By this method, the search complexity is simplified and computation is reduced.

3.3.PROVIDING QOS ROUTING

In order to meet different network requirements, we add the QOS to NCOF. As we know, we can add MPLS label to packet header to transmit the coding information (Figure 3 and Figure 4). It gives us an enlightenment to apply MPLS label to QOS. Actually, it has already been realized called MPLS QOS. But when we applied MPLS QOS to NCOF, we should make appropriate changes (Figure 5). We add a priority number to packet header (Figure 3 and Figure 4) so that the coding queue can be further divided according to this priority. In addition, the waiting time of the coding packet is different depend on the different priority queue.

4.IMPLEMENTATION AND EXPERIMENTS

We now present our framework that is guided by the design goals described in Section III . Firstly, we created a set of services of MMF-NC routing on the OpenFlow controller. For the OpenFlow switches, new actions about NC should be defined to perform the buffers, encoding and decoding functions.

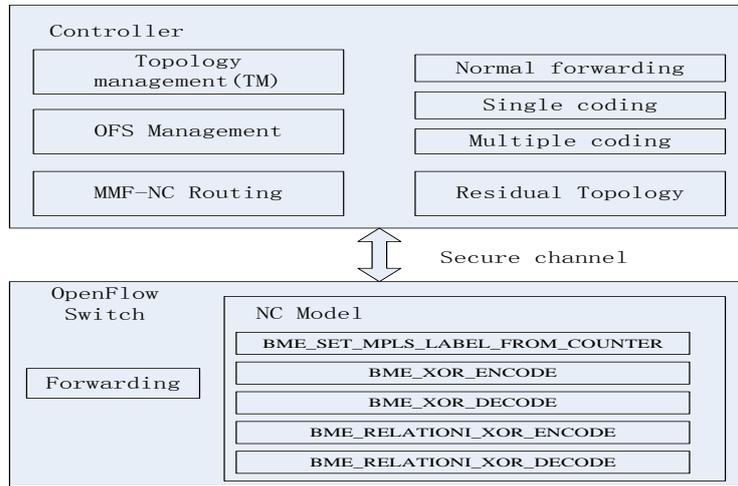


Figure 6. The NC-OF framework

From Figure 6 we can see the NC-OF framework, the basis modules we add in controller are Topology Management(TM), OpenFlow Switch(OFS) Management, MMF-NC Routing, Residual Topology and three different experimental forwarding methods. TM discovers and maintains network connectivity through controller-to-switch messages and asynchronous messages; OFS Management detects congested links by collecting link utilization counted per port on the OFS periodically; MMF-NC Routing calculates NC routing paths and the Residual Topology is in the service of MMF-NC Routing modules. To support NC actions on OFS, we extended NC model which includes five actions in OpenFlow switch as shown in table 1. The OFS receive message from controller and install routing into routing tables, forward the packets as routing tables.

Table 1. Added Actions in OpenvSwitch

ACTION	USAGE
BME_SET_MPLS_LABEL_FROM_COUNTER	Make sequence number for packets;
BME_XOR_ENCODE	Parsing coding information; Stored into the encoding queue;
BME_XOR_DECODE	Parsing coding information; Stored into the decoding queue;
BME_RELATIONI_XOR_ENCODE	Parsing encoding relations and the coding information; Stored in the encoding queue;
BME_RELATIONI_XOR_DECODE	Parsing decoding relations and the coding information; Stored in the decoding queue;

We test NC-OF on Mininet, one of the popular network simulation platforms. In order to observe the data flow transmission process, we design a traffic monitor in Mininet . When no flow in the network, the line between switches is displayed as blue, and when there are flows, the line is displayed as red, as shown in Figure 7.

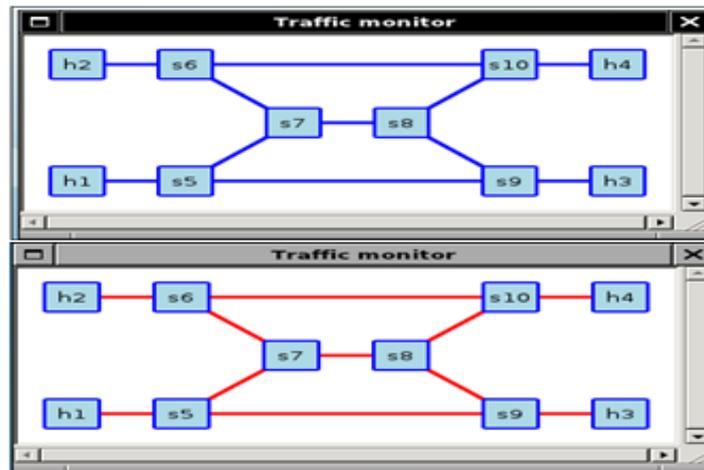


Figure 7. Traffic monitor

Figure 8 is our test topology, we didn't use large network topology because we want to focus on the process of network coding, try to ignore the influence of the intermediate node forwards.

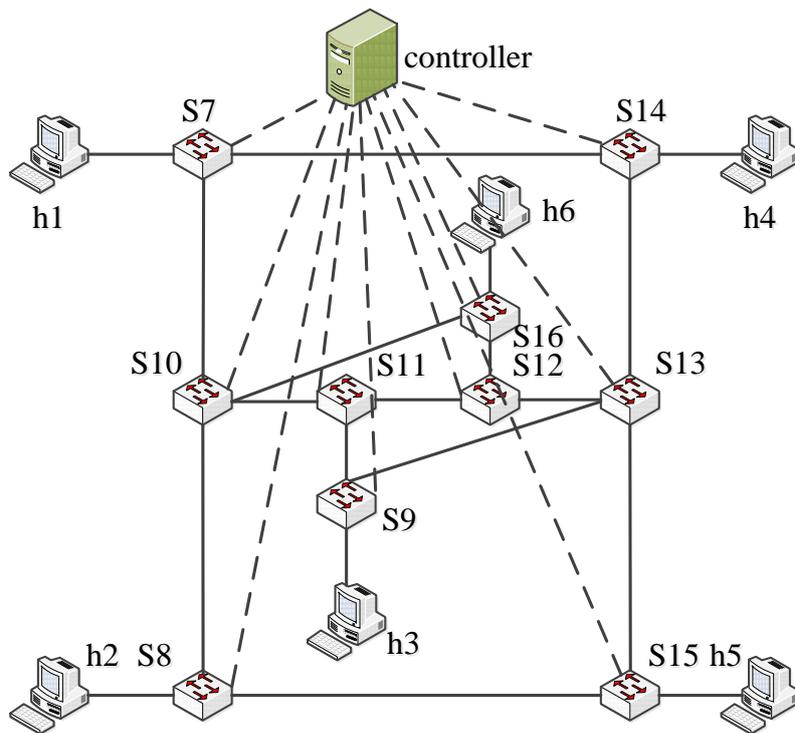


Figure 8. test topology

As Figure 8 shows, there are three flows (the size of the packet in the flow is the same) in the network , respectively are flow1 from client h1 to h5, flow 2 from client h2 to h4, flow 3 from client h3 to h6. Then setting the switch port rate is 0.96Mbit/s. we tested the time delay under different forwarding strategy. We can see the three forwarding strategy in Figure 9, the first one is normal forwarding, the second one is for single coding and the third one is for multiple coding .

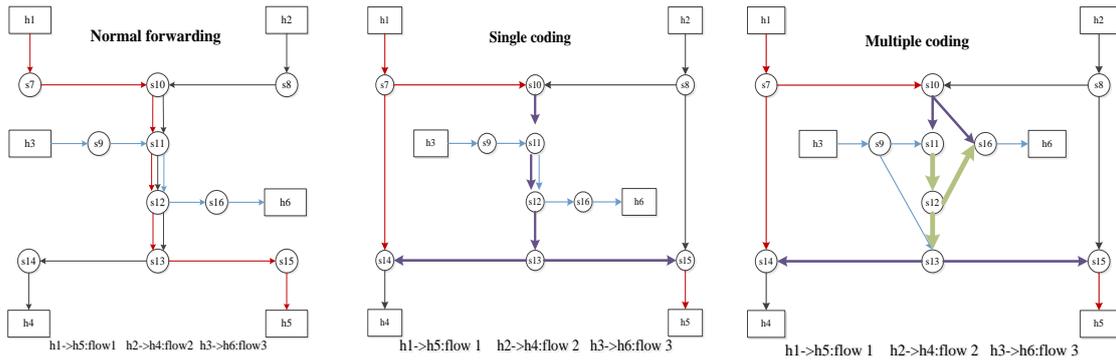


Figure 9. Different forwarding strategy

The experimental results as shown in Figure 10, in normal forwarding, when the transmission load is greater than 320kb/s, the average delay of flows showed a substantial upward trend, mainly due to the port transmission rate is less than the reception rate of the switch. So the link between s10 and s11, s11 and s12, s12 and s13 become congestion. Similarly, when the transmission load exceeded 480kb/s in single coding, it also appears link congestion lead to time delay. While multiple coding improved the link capacity through coding several times has not appeared above two cases.

But through the experimental results, it can be seen that the average transmission delay of multiple coding and one coded transmission delay is higher than the normal transmission delay when the link bandwidth is enough. It mainly due to the coding strategy needs to encode the data packets, increases the transmission delay. Therefore, although the network coding can improve link capacity, but also can bring problem of time delay, the increase in the amount of calculation and so on. It has both advantages and disadvantages.

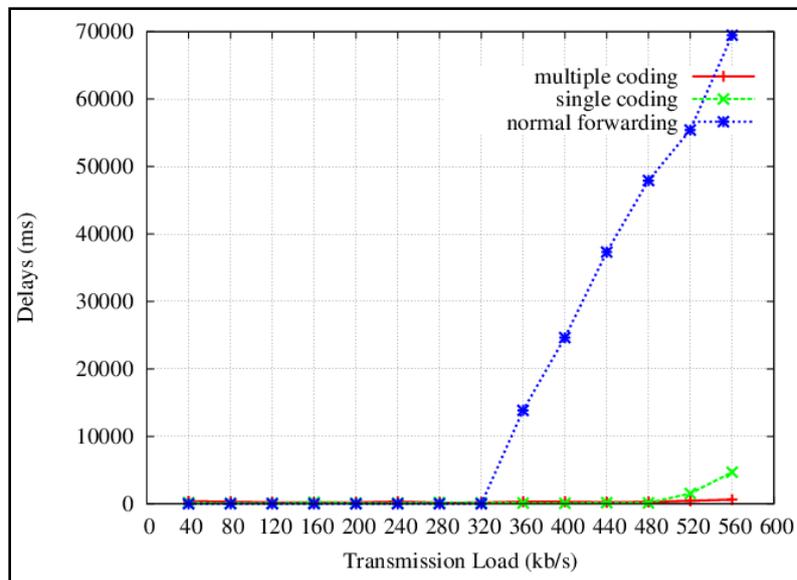


Figure 10. The flows transmission delay change with the transmission load

In addition to test each strategy flow average delay along with the change of bandwidth, we also did a survey about packet receiving rate with the change of transmission load in three forwarding strategy.

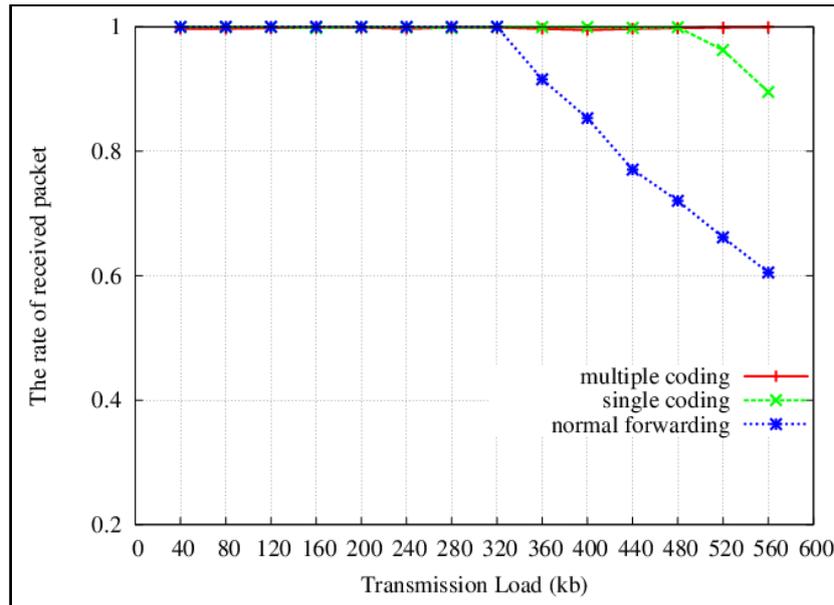


Figure 11. the rate of receiving packet with the change of transmission load

The result is shown in Figure 11, when the bandwidth is less than the demand, the packet will be lost. This is because the network link congestion induced packet needs to staying in OpenFlow switch and waiting to be sent. When the packets waiting time exceed a certain limit, the switches will lost the packets, which leads to the rate of receiving packet decreased

5.CONCLUSIONS AND FUTURE WORK

This paper focus on the problem in implementing Network Coding in the OpenFlow architecture. We put forward the relevant technical challenges, give effective solutions, and design the NC-OF framework in order to verify the reliability of solutions. We implemented NC-OF framework on Mininet to evaluate its performance. However, NC-OF still has some problems need to solve, such as adapting to the change of network topology with low delay, meeting different QOS requirements and so on. We will try to solve these problems in the future.

ACKNOWLEDGEMENT

This work was supported by the National Key Technology Research and Development Program of the Ministry of Science and Technology of China under Grant no.2012BAH93F01, the Innovation Research Fund of Huazhong University of Science and Technology, no.2014TS095, the National Science Foundation of China under Grant no. 60803005

REFERENCES

- [1] Xiaowen Chu, 'Random linear network coding for peer-to-peer applications', Network, IEEE, July-August 2010
- [2] Dimakis, A.G. Dept. of Electr. Eng, Univ. of Southern California, Los Angeles, CA, USA Ramchandran, K.; Wu, Y; Changho Suh 'A Survey on Network Codes for Distributed Storage', Proceedings of the IEEE, March 2011
- [3] Stanford University. About Clean Slate [EB/OL]. http://cleanslate.stanford.edu/about_cleanslate.php, 2012-5-30.
- [4] N. McKeown et al, "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM CCR, 38(2), 2008.

