Exploit Collision: Network Coding in Switch based on VOQ Scheduling Algorithm

Li Ma, Hui Li, Shijie Lv, Guanghui Zhang and Fuxing Chen

***Abstract*—To improve the performance of network and reduce implementation cost, many switching fabrics have been proposed, among which Shared Bus, Shared Memory, Crossbar and Multipath Self-routing Switching Structure (MSSS) are especially noteworthy. The first three are not suitable for large-scale expansion because of bandwidth bottleneck or inefficient ability to schedule I/O match. MSSS is featured by the properties of high modularity, self-routing, low component complexity with no internal buffer and no buffer delay and jitter. However, it may incur internal collision, thus improvement is needed for practical application. In this paper we exploit collision by contacting Network Coding (NC) with MSSS, thus achieving the increase throughput in network when the traffic load is relatively heavy. On this basis, to solve the head of line blocking (HOL) problem, we add virtual output queue (VOQ) scheduling algorithm to allow NC play better results. We implement this design on FPGA, and the results indicate that this structure has effectively reduced the packet loss rate.**

***Index Terms*—collision，Network Coding， switch， VOQ**

I. Introduction

ALONG with the development of Internet, dramatic increase in the number of user has brought great challenges to network, which is composed of transmission lines and switching routers. The ability of network depends on the bandwidth of router and line card. However, the development of router is far behind transmission bandwidth as it is poorly satisfied people's need by the current supply of bandwidth. To this case, various packet switching structures have been proposed to broadband network, such as Shared Bus [1], Shared Memory, Crossbar with combined input and output queue, etc.

For Shared Bus switch, the switch size is limited by the memory read/write access speed. Although Shared Memory switch has good performance in buffer utilization and delay [2], it is restricted due to memory speed constraint when the line speed or the port size increases. Crossbar switch has three attractive properties: internally non-blocking, simple and modular [3], which makes it an ideal structure when the port size N is small. Since the number of cross-point is N^2, however the complexity is as high as $O(N \log N)$ when constructing a large multistage switch with small Crossbar switch, thus introducing the unacceptable cost when N is large [4].

To construct a structure for large scale switching routers, He Wei et al. proposed a model named Multipath Self-routing Switching Structure (MSSS) [6], which plays an irreplaceable role in correctly and efficiently switching packets without internal buffer, buffered delay and jitter. But its inevitable internal collision has limited its application.

Here we proposed a model, which combines Network Coding (NC) with MSSS to further exploit the advantage of MSSS. NC has been widely used in many network communications, which is firstly proposed by Robert Li [5]. The lost information will be recovered by encoding and decoding to improve the communication reliability.

Apart from that, we add virtual output queue (VOQ) [8] scheduling algorithm in this model to better implement NC and switching. When several packets from different input ports are destined for the same destination simultaneously, collision then will come out. To solve this problem, a variety of VOQ scheduling algorithms have been proposed such as Iterative Round-Robin with SLIP (iSLIP) [6], Ping Pong Arbiter (PPA) [7], which can be generally classified into two categories: centralized and distributed. The primary difference among these various algorithms is how to make a connection between the inputs and outputs. But the majority of them devote themselves to make a conflict-free match to connect the inputs with outputs.

In our architecture, we combine the three factors above and take full advantage of the inter-contention while inter-contention is a nerve-wracking problem in other models. Moderate packet loss and inter-collision are allowed because NC can help to recover the imperfect packet to a certain degree. VOQ here can avoid the head of line blocking (HOL). In addition, it can help to implement NC more convenient.

The rest of this paper is organized as follows: Section II gives an overview of the whole system. Section III describes the theoretical analysis about the packet loss rate and simulation result; and Section IV presents how to design and implement this kind of switch based on NC and Multipath Self-routing Switching Structure; The performance of the proposed switching architecture is analyzed and discussed in Section V. Finally, we come to a conclusion based on the system model mentioned above in Section VI.

Manuscript received January 8, 2015. This work is supported by National Basic Research Program of China (973 Program, No.2012CB315904), the National Natural Science Foundation of China (No.NSFC61179028), the Natural Science Foundation of Guangdong (GDNSF, No.S2013020012822), and the Basic Research of Shenzhen (No.JCYJ20130331144502026, No.JCYJ20140417144423192).

Li Ma, Hui Li, Shijie Lv, Guanghui Zhang, Fuxing Chen are with the Shenzhen Engineering Lab of Converged Networks Technology, Inst. of Big Data Technology, Shenzhen Graduate School, Peking University, Shenzhen, Guangdong, 518055, China (email: mali5057@163.com, lih64@pkusz.edu.cn, lvshijie217@163.com, zhangghabc@163.com, chenfuxing09@vip.qq.com)

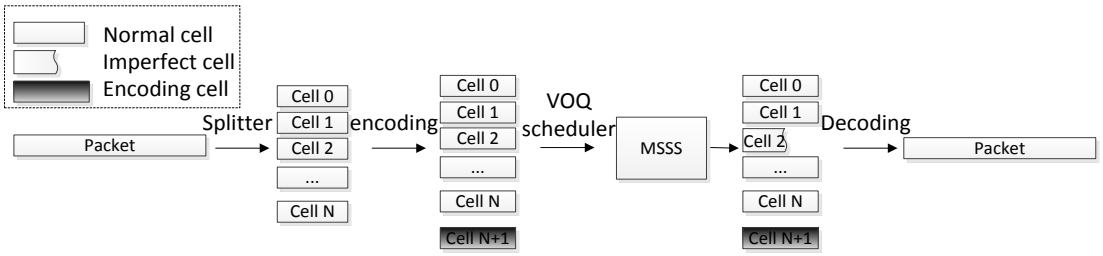


Fig. 1 Packet processing in the NC based switching system

II. MODELING AND THEORETICAL BASIS

A. Overview of System Model

The whole system is shown in Fig.2. For example, one packet from Input 1 is destined for one of the outputs. Firstly it will be split into *N* cells with fixed length by Splitter. Then the Network Coding cell is generated based on the information of the previous *N* cells in Encoder. At last, all the *N*+1 cells are sent to the MSSS via VOQ. Note that cells may be lost during the above process due to various conditions such as collisions in the MSSS or a full buffer. If there is only one cell in the packet that is lost, then we can recover the original packet with the encoding cell in Decoder. The detail about packet processing is shown in Fig. 1. As shown in the graph, the third cell (Cell 2) has lost partial data and the NC system can still recover the original data by decoding.

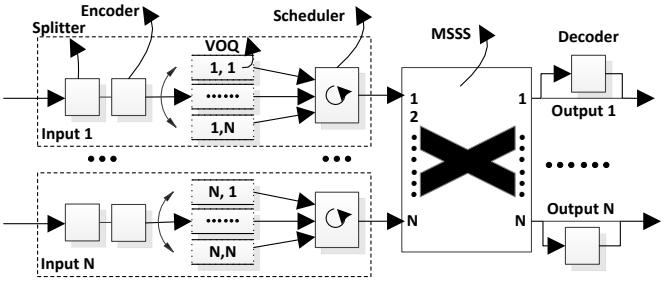


Fig. 2 Overview of NC system

B. Multipath Self-routing Switching Structure

MSSS is characterized by self-routing [10] feature, which consists of Multistage Interconnection Network (MIN) and 2G-to-G Concentrator. Fig. 3 illustrates a MSSS based on MIN and 16-to-8 Concentrator. In this structure, parameter *G* represents the group size, *M* indicates the quantity of groups, and *N* is the port size [9], which is equal to $M \times G$.

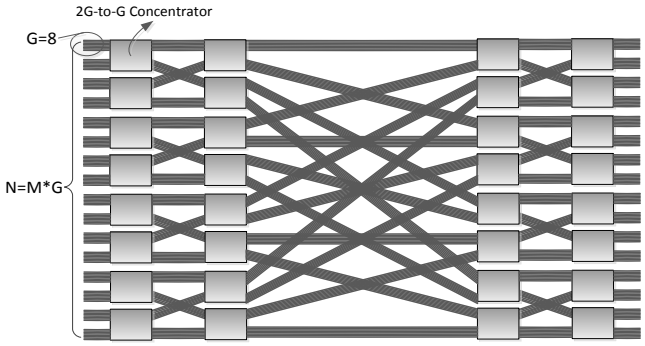


Fig. 3 Multipath Self-routing Switching Structure
(*M*=16, *G*=8)

C. Network Coding Engine

Network Coding Engine contains two modules: Encoder and Decoder. For example, the Encoder module performs an XOR operation among these cells to generate a network-coded cell, as follows:

$$cell\_N+1 = cell\_1 \oplus cell\_2 \oplus ... \oplus cell\_N \quad (1)$$

Then all the cells are sent to the MSSS. The Decoder module receives cells from MSSS and assembles these cells into the original packet. If there is a cell (for example *cell_n*, not the encoded cell) of the packet is lost in the switch, the Decoder module can still recover the cell just XOR all the received cells together, as shown in the following:

$$\begin{aligned} cell\_n = {} & cell\_1 \oplus cell\_2 \oplus ... \oplus cell\_n-1 \\ & \oplus cell\_n+1 \oplus \ldots \oplus cell\_N+1 \end{aligned} \quad (2)$$

III. THEORETICAL ANALYSIS

A. Packet Loss Rate Calculation

In this section, we will analyze the packet loss rate about the system model. The arrived packet of each input port is assumed to be identical independently distributed. As mentioned above, firstly we divide a packet into cells and then send these cells to switching fabric. Therefore, we further assume the length of a packet is uniformly distributed. For a standard Ethernet packet (length between 64 bytes to 1518 bytes), the number of cells are between 1 and 12 when we set the cell length is 128 bytes. To simplify the computation, here we ignore the influence of VOQ algorithm on the performance evaluation.

Assume the cell blocking probability is P_c, therefore, for a packet within n cells, the packet loss rate P_p can be calculated as follows:

$$P_p = 1-(1-P_c)^n, 1 <= n <= N \quad (3)$$

Where *n* is the number of cell in the packet and *N* is the maximum number of cell in the packet and here *N* is equal to 12.

For system without network coding, the packet loss rate P_L is

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

$$P_L = \sum_{n=1}^{N} \frac{1}{N}[1-(1-P_{c1})^n], 1 <= n <= N \quad (4)$$

For system with network coding, the packet loss rate P_L' turns out to be

$$P_L' = \frac{P_{c2}}{N} + \sum_{n=3}^{N} \frac{1}{N}[1-(1-P_{c2})^n], 1 <= n <= N \quad (5)$$

Where P_{c1} and P_{c2} represent cell blocking rate of two different systems respectively. And the following formula will show how to calculate them.

As mentioned above, for a 4×4 switching fabric with two-stage composed of concentrators, assume the group size is *G*, the phenomenon of cell loss only happens when there are more than *G* cells with the same destination address sent into the Concentrator. Suppose the loading rate for each line of input is *p*, we can obtain the following formula to calculate the cell loss probability P_{s1} and P_{s2} for the concentrator of stage one and stage two, respectively:

$$P_{s1} = \sum_{k=G}^{2G-1} C_{2G-1}^{k} p^k (1-p)^{2G-1-k} \cdot \left(\sum_{j=G}^{k} \frac{j-G+1}{j+1} C_k^j (1/3)^j (2/3)^{k-j} \left(\sum_{s=1}^{j} C_j^s (1/4)^s \right) \right) \quad (6)$$

$$P_{s2} = \sum_{k=G}^{2G-1} C_{2G-1}^{k} p^k (1-p)^{2G-1-k} \cdot \left(\sum_{j=G}^{k} \frac{j-G+1}{j+1} C_k^j (1/2)^j (1/2)^{k-j} \right) \quad (7)$$

Thus, it's obvious to know that $P_{c1} = P_{s1} + P_{s2}$, where P_{s1} is the cell loss probability considering the internal conflict of the first stage and P_{s2} is the cell loss probability for external contention of the second stage.

Analogically, we can derive the cell loss probability for a concentrator with network coding, as follows:

$$P_{ncs1} = \sum_{k=G}^{2G-1} C_{2G-1}^{k} p^k (1-p)^{2G-1-k} \cdot \left(\sum_{j=G}^{k} \frac{j-G}{j+1} C_k^j (1/3)^j (2/3)^{k-j} \left(\sum_{s=1}^{j} C_j^s (1/4)^s \right) \right) \quad (8)$$

$$P_{ncs2} = \sum_{k=G}^{2G-1} C_{2G-1}^{k} p^k (1-p)^{2G-1-k} \cdot \left(\sum_{j=G}^{k} \frac{j-G}{j+1} C_k^j (1/2)^j (1/2)^{k-j} \right) \quad (9)$$

$$P_{c2} = P_{ncs1} + P_{ncs2} \quad (10)$$

B. Performance Analysis and Simulation

For a better understanding of the above, we made a simulation with MATLAB by plotting the packet loss rate as the function of loading rate p in Fig. 4. From the figure, we can find that the Network Coding system has an improvement compared with non-NC system performance. When the packet sending rate is small (less than 40%), the gap between two curves is larger than 10dB. That is, compared with traditional switching fabric, the system with NC can obtain a much more benefit when the sending rate is low.

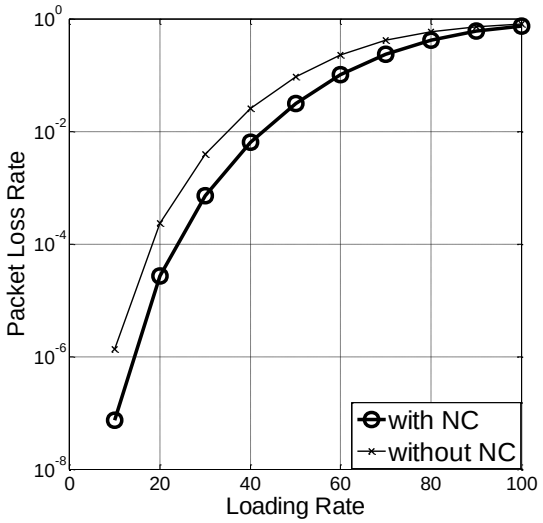


Fig. 4 Packet loss rate for a four-by-four switching fabric

IV. SYSTEM DESIGN AND IMPLEMENTATION

The NC system model has been implemented on Altera Stratix V FPGA, which can offer the transmission speed as fast as 10Gbps. Here we developed the system on four by four MSSS (*G*=8, *K*=4).

The whole system is shown as Fig. 5, which mainly contains three layers: Physical layer (10G PHY), MAC layer (10G MAC) and user-defined layer (user_data_path). Mac_adapter offers an interface from MAC layer to user-defined layer. We can classify the whole system into two data channels: input/output data channel as shown in the blue and red arrows, and debug data channel with black arrow. The former is for transmission and processing of data, while the latter is used to debug by extracting data from sub-modules. When the host wants to access information from sub-modules, it will send a request with address to the bus. Then the request will be sent to submodules by bus. At last, the sub-module who has the same address will make a response to the request.

A. User Data Path (UDP) Design

UDP is the key module of the whole system, which focuses on the data processing and mainly includes Splitter, Encoding, VOQ, MSSS and Decoding. It completes most work in the data flow: adding head, slicing a packet into different cells, performing a network encoding among cells, managing a virtual output queue, switching cells according to their destination address and assembling and decoding cells to generate the original packet.

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

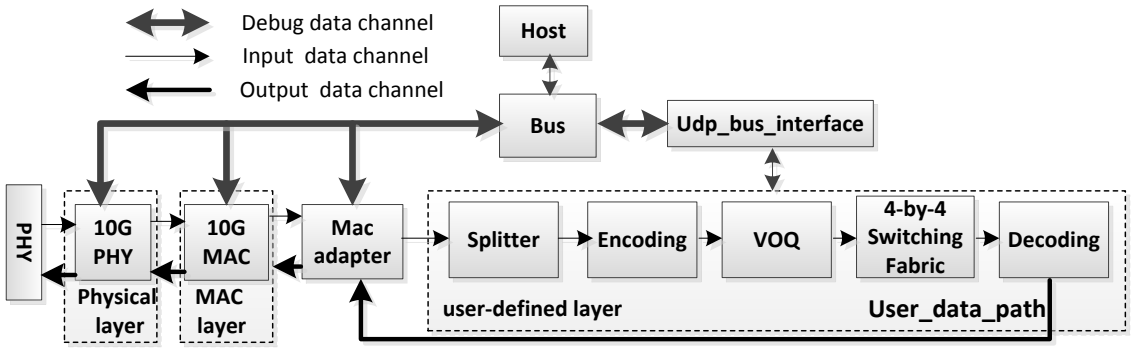


Fig. 5 The whole system design

1) *Encoding*: carry out a encoding function among these cells;
2) *VOQ*: implement a simple distributed algorithm to push cells to switching fabric;
3) *Four by four MSSS*: a standard Multipath Self-routing Switching Structure with group size G 8;
4) *Decoding*: completes the task of decoding a lost cell (if necessary) and assembling cells into a packet.

B. Format of Cell Design

In order to transform the normal cells and encoding cell to switch, a header with useful information is required. The format of cell is shown by Fig. 6.

1) *Regular head*: it indicates the control bits for switching and other useful information.
2) *NC indicator*: flag that indicates whether the cell is a network-coded cell.
3) *Number of cells*: how many cells in the packet.
4) *Cell ID*: the sequence number of the current cell in the packet.
5) *Packet ID*: it is used to distinguish cells among different packets.

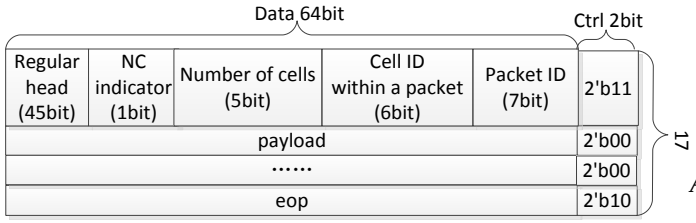


Fig. 6 The format of a cell

Cell consists mainly of two parts, 64bit data and 2bit control information, which plays a role of synchronization: 2'b11 represents the start of cell, and 2'b 10 indicates the end of cell while 2'b 00 represents payload of cell.

C. Description of VOQ Algorithm

To better implement NC in MSSS, we add a distributed VOQ scheduling algorithm, providing a fast and fair arbiter design depending on the traffic load rate.We classify the state of traffic model in each port into three categories: HEAVY, FAIR and LIGHT. Different measures are made for different circumstances. Here we consider a switch of four by four ports.

Step 1: judge the state of *VOQ*.

Definition 1: ε_L is the flag factor of state "light" for *VOQ*; ε_H is the flag factor of state "heavy"; $\varepsilon(i, j)$ is the factor for state of $VOQ(i, j)$, which is the occupancy rate of $VOQ(i, j)$;

1) If $0 < \varepsilon(i, j) < \varepsilon_L$, we define the state of as "light";
2) If $\varepsilon_L \le \varepsilon(i, j) \le \varepsilon_H$, we define the state of as "fair";
3) If $\varepsilon(i, j) > \varepsilon_H$, the state is heavy.

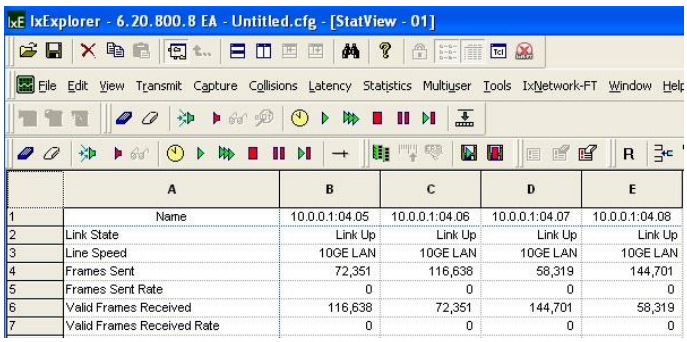
Step 2: judge the state of traffic model of each port.
If there are at least three VOQs being heavy state, the state of traffic model in this port is HEAVY; If there are at least three VOQs being light state, the state of traffic model in this port is LIGHT; Otherwise, it will be FAIR.

Step3: make measures: choose cells from the VOQs according to the state of traffic model.

1) HEAVY: 6 cells are selected from the heaviest VOQ, 2 cells from the second heaviest VOQ;
2) FAIR: 4 cells are selected from the heaviest VOQ with another 4 cells from the second heaviest VOQ;
3) LIGHT: 2 cells are selected in each VOQ. The VOQ whose valid cell number is less than 2 will send 1 or zero cell.

V. System Testing and Performance Analysis

A. System Testing with Real Network Traffic



	A	B	C	D	E
1	Name	10.0.0.1:04.05	10.0.0.1:04.06	10.0.0.1:04.07	10.0.0.1:04.08
2	Link State	Link Up	Link Up	Link Up	Link Up
3	Line Speed	10GE LAN	10GE LAN	10GE LAN	10GE LAN
4	Frames Sent	72,351	116,638	58,319	144,701
5	Frames Sent Rate	0	0	0	0
6	Valid Frames Received	116,638	72,351	144,701	58,319
7	Valid Frames Received Rate	0	0	0	0

Fig. 7 Test result with IXIA
(10.0.0.1:04.05~10.0.0.1:04.08 represent Port 0 to Port 3)

After imlementation, we carried out various experiments under real network traffic to test and verify the performance

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

on platform IXIA, which is a network test instrument to generate or capture standard Ethernet frames transmitted at the rate of 10/100/1000 /10000Mb/s. As shown in Fig. 7, we can see that when Port 0 and 1 send and receive data packets to each other, there are no data dropped in the outputs. The situation is same for Port 2 and 3.

B. Performance Analysis

Fig. 8 shows the performance comparison among the NC and non-NC systems: the fine curves are for the system without NC and the coarse ones for the system with NC. The packet loss rate of the NC system is obviously less than that of non-NC system. Some of the modules in the system become saturated when the loading rate is larger than 0.6.

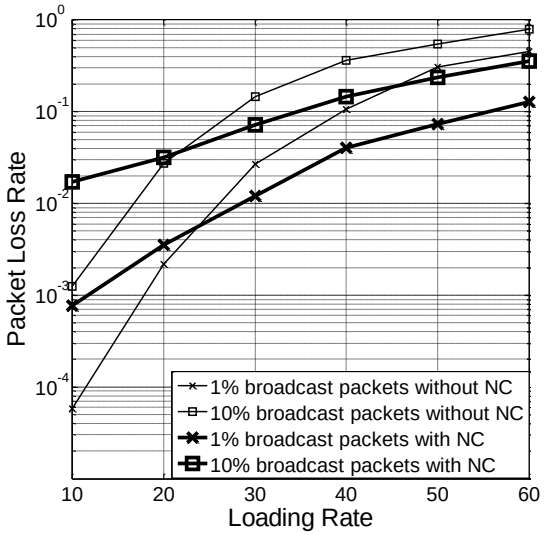


Fig. 8 Performance of NC and non-NC

VI. CONCLUSION

In this paper, we implement the design contacting NC system based on VOQ on DE 5 FPGA. This system is compatible with the 10Gbps Ethernet. With NC, the packet loss rate of the switching system can be decreased 90%, compared with the traditional MSSS system when the traffic load is relatively heavy. This can greatly reduce the retransmission in network and improve the network throughput. Going forward, we plan to add a feedback mechanism to reduce the packet loss rate to zero.

ACKNOWLEDGMENT

The author would like to thank the help of Minglong Zhang and Kai Pan for proofreading the draft.

REFERENCES

[1] Pawloski M B. Shared bus in-circuit emulator system and method: U.S. Patent 5,313,618[P]. 1994-5-17.
[2] Andrade P, Cooperman M, Sieber R W. ATM shared memory switch with content addressing: U.S. Patent 5,513,134[P]. 1996-4-30.
[3] Javidi T, Magill R, Hrabik T. A high-throughput scheduling algorithm for a buffered crossbar switch fabric[C]//Communications, 2001. ICC 2001. IEEE International Conference on. IEEE, 2001, 5: 1586-1591.
[4] Yoshigoe K, Christensen K J. A parallel-polled virtual output queued switch with a buffered crossbar[C]//High Performance Switching and Routing, 2001 IEEE Workshop on. IEEE, 2001: 271-275.
[5] Li S Y R, Yeung R W, Cai N. Linear network coding[J]. Information Theory, IEEE Transactions on, 2003, 49(2): 371-381.
[6] N. McKeown, "iSLIP: a scheduling algorithm for input-queued switches," IEEE/ACM Trans-actions on Networking, vol. 7, no. 2, pp. 188–201 (Apr. 1999).
[7] Chao H J, Lam C H, Guo X. A fast arbitration scheme for terabit packet switches[C]//Global Telecommunications Conference, 1999. GLOBECOM'99. IEEE, 1999, 2: 1236-1243.
[8] Banovic D, Radusinovic I. Scheduling algorithm for VOQ switches[J]. AEU-International Journal of Electronics and Communications, 2008, 62(6): 455-458.
[9] Hui Li, Wei He, Xi CHEN, Peng Yi, Binqiang Wang, "Multi-path Self-routing Switching Structure by Interconnection of Multistage Sorting Concentrators", IEEE CHINACOM2007, Aug.2007,Shanghai;
[10] F.X. Chen, Hui LI, etc, "The Wire-speed Multicast Switch Fabric Based on Distributive Lattice", IEICE Transactions on Communication, 2014.