A Distributed Scheduling Algorithm in Central-Stage Buffered Multi-stage Switching Fabrics

Yuxiang Hu, Fang Dong, and Julong Lan

National Digital Switching System Engineering & Technological Research Center
Zhenzhou, Henan, P.R. China, 450002
{huyuxiang1982,chxachxa}@yahoo.com.cn

Abstract. The current MSM switching fabric has poor performance under unbalanced traffic. To eliminate the internal congestion of switching fabric, we put forward a new central-stage buffered multi-stage switching fabric—CB-3Clos and the backpressure-based strategy to control flows under credit-dispensed mode. By analyzing the condition to satisfy the central-stage load balance, we also advance an iSLIP alike scheduling algorithm—RGA. The simulation results show: compared with CRRD algorithm based on MSM switching fabric, the RGA algorithm has high throughput irrespective with the arriving traffic model and better performance in packet delay. At the same time, the QoS can be guaranteed.

1 Introduction

Current network faces the embarrassment of unbalance between transport ability and switching ability. Compared with the transport ability developing at a fast speed, the lag of switching ability of core nodes has became the bottleneck of network. The single-stage switches and the relevant scheduling algorithms have became mature, so it's difficult to increase the number of ports or heighten the line-rate to satisfy the requirements of large-scale switching system, the single-stage switch meets it's bottleneck. At present, in order to achieve large-capability and high-expansibility, the development shows an obvious trend: multi-stage switching fabric. Its purpose is taking advantages of multi-stage switching fabrics to design switch systems with large-capability and high-expansibility.

Current researches on switching fabrics mostly focus on Clos network, Benes network, Banyan network and so on. In all of these networks, the three-stage Clos network has become the emphasis owing to its high-modularization and large-capability and the characteristic of strict non-blocking in internal links.

The three-stage symmetrical Clos network C(n, m, r) has r n×m switch cells in the input-stage, and m r×r switch cells in the central-stage, so there are r m×n switch cells in the output-stage. The network has N=n×n input/output ports in all, and every switch cell in central-stage connects with every switch cell both in input-stage and output-stage through a link. It has been proved that [1]: if $m \geq n$, the C(n, m, r) Clos network is a rearrangeable non-blocking switching fabric, this indicates that if and only if there are a pair of matched input/output ports and a pair of unmatched

M. Xu et al. (Eds.): APPT 2007, LNCS 4847, pp. 121–130, 2007.

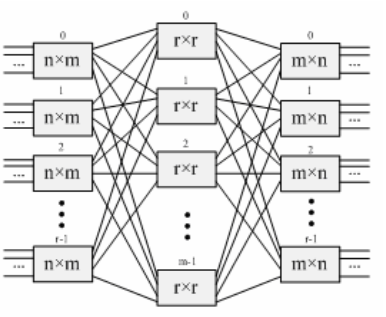

Fig. 1. The three-stage symmetrical Clos network C(n, m, r)

input/output ports, by re-routing between the matched ports, we can establish a match between the pair of unmatched ports. Figure 1 shows the three-stage symmetrical Clos network C(n, m, r).

2 Related Work

2.1 Buffer Setting

The Clos network in pocket switching system falls into two categories: the first is SSS Clos fabric [2]. This fabric takes advantages of space-division multiplexing and makes the bufferless Crossbar as the switch cell. The SSS Clos fabric is simple in physics and has loose requirements on circuit level, but it's strict with the scheduling algorithms and makes them difficult to implement.

With the development of circuit level, buffers have been applied into multi-stage fabric, so the MSM Clos fabric [3] became feasible. The MSM Clos fabric contains buffers in the first and last stage, but use bufferless central-stage. It has been proved that: under uniform traffic, the CRRD algorithm based on MSM can provide nearly 100% throughput, but the performance drops dramatically under unbalanced traffic.

2.2 Flow Control

2.2.1 Backpressure

Backpressure protocol provides a direct method to prevent data from overflow at input ports. It's similar to the counterpressure brought by flows in the pipeline, when the end is closed, the flows will generate counterpressure to the source, and so interdict or slow down the flows. Similarly, the congested ports will send the congestion messages to the sources and make the sources restrict pockets from being sent into network.

Backpressure technique can be selective applied on certain logical links; it's convenient to manage these links between two nodes. The backpressure-based flow control is a good mechanism in the case of whole net, and can be applied in network infrastructures that allow hop-by-hop flows such as the routers, but is restricted in this.

2.2.2 Credit-Based Flow Control

Another universal mechanism is credit-based flow control [5]. Its essence: before send any packet, the sources must receive the credit messages from receivers and the credits decide the number of packets can be sent. Generally, the credit equals to the line-rate multiplying the RTT (round-trip time).

Ideally, the credit-based flow control mechanism can make sure no packet lost, even if under burst traffic, because the length of packet queue will not exceed the credit. Due to this strategy would keep a queue for per link and makes the buffer bigger, it's not suitable for multi-link, and otherwise the buffer will be very complex. The credit-based flow control is usually implemented in low-cost adapters and can achieve high performance.

2.2.3 Regional Explicit Congestion Notification (RECN)

To resolve the internal blocking in multi-stage switching fabric, it's pivotal to resolve the sharing queues between congested and non-congested flows while ensuring the performance. J. Duato and I. Johnson put forward an extensible congestion control strategy [6]. They advise: all non-congested flows share a single queue, while dynamically assign a set-aside-queue for per congestion tree. Congestion trees may be rooted at any output or internal fabric link, and their appearance is signaled upstream via regional explicit congestion notification (RECN) messages. This queue-sharing method based on space-division multiplexing resolves the effects between congested flows, while the complexity of the algorithm is high. The packet delay generated by the transport of the signals is also another problem.

3 Scheduling Algorithm in Multi-stage Switching Fabrics

3.1 Buffer Assignment

The switching fabric may be buffered or bufferless. The bufferless switching fabrics merely steer the flows and couldn't generate any packet delay or output congestion, but it makes the scheduling algorithms very complex and impractical. The buffered switching fabric can resolve the port collision to some extent by setting buffers in the fabric, and current circuit level makes this fabric feasible. Buffer has become an important factor in deciding the performance of the switching fabrics.

Based on the analysis above, inheriting the thinking of space-division multiplexing in multi-stage fabrics, and using the single-stage switches for reference, we put forward a new central-stage buffered three-stage Clos switching fabric—CB-3Clos, figure 1 shows the configuration.

As shown in figure 2, VOQs are set in the input-ports of the switch cells at input-stage to store the congested flows. Here we use the two-stage switching for reference and set buffers in the input-ports of the switch cells at central-stage. So the first and second stage can be treated as a two-stage switching. For the sake of internal non-blocking, the fabric introduces multi-route. Per traffic is allotted to all switch cells of the central-stage uniformly, in a way such as to equalize the rates of the resulting sub-flows. The central-stage switch cells can be thought of as parallel slices of one, faster virtual switch, and inverse multiplexing performs load balancing among these slices.

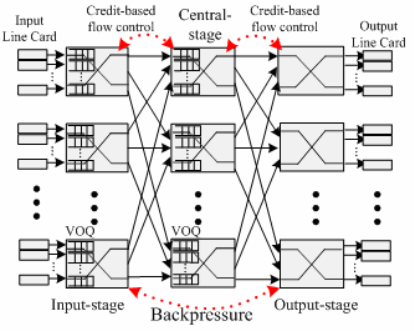

Fig. 2. The configuration of CB-3Clos switching fabric

By setting buffers at the central-stage, the complexity of algorithms on the first stage is reduced efficiently, and it can decrease the unfairness of resources at central-stage brought by the unbalanced dispersed packets' arriving from input-stage. So, to some extent, the load balance on the first two stages can tidy the traffic—tidying the unbalanced traffic into regulated traffic.

3.2 Congestion Control

The scheduler of switch is a buffer-disperser for packets in essence, and the scheduling strategy is just a traffic control strategy selected to guarantee the performance. Considering current traffic control strategies, based on the configuration of CB-3Clos, we put forward a strategy to control traffic which is based on backpressure protocol while in credit-dispensed mode: use backpressure protocol in interior of the whole fabric but in credit-dispersed mode between two conjoint stages. As the broken lines shown in figure 2, firstly the input ports of input-stage send requests to their corresponding ports duo to the queue's estate, secondly the switch cells of output-stage produce credits by calculating the queues and the buffers, and then send credits to central-stage, after receive the credits, the switch cells of central-stage send them to input-stage by load balance. Only the input ports that have obtained credits can send packets into switching fabric.

Traffic first being store in buffers then entering fabric, this can obtain high throughput and small buffer in interior of the fabric. It's a typical thinking of space-division multiplexing, and it makes only the traffic destined out can occupy the buffer, while the congested traffic stay out of fabric. At the same time, to make full use of the buffer which is very costly, the buffer is reserved from the output stage to input stage, once a stage. This is precisely opposite to how packets progress under backpressure protocol. The direction chosen ensures that each reservation, when performed, is on behalf of a packet that is guaranteed not to block inside the buffer: buffer space has already been reserved for that packet in the next downstream buffer. Hence, packets will be allowed to move freely, and without danger of any buffer overflowing. Figure 3 gives a demonstration of the congestion control.

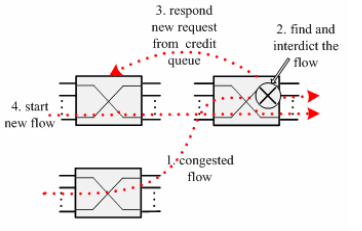

Fig. 3. Demonstration of the congestion control in interior of the fabric

This strategy may lead to the phenomenon that several ports reserve buffers for the same packet; here we define this phenomenon as output buffer collision. The output buffer collision may lead to: buffer reservations constitute a second pass through the fabric, after requests have traversed once from inputs to the per-output scheduler. So the most essential method is making sure that no other input ports send any request to these output ports.

In sum, we give the rules to resolve the congestion between flows: in per input/output port, per credit scheduler work paralleling in pipe-line independently; the request must fix on certain output port for per flow and be sent to corresponding port; requests must queue in the credit scheduler, per output scheduler responds to these requests after assigns buffers. Credits may be generated based on certain QoS strategies such as: WRR/ WFQ. When packet comes out of the fabric, it must inform the credit scheduler to re-assign buffer, so the department rate can regulate the rate that switch be granted.

3.3 Load Balance in Central-Stage

In multi-path fabrics, route can be selected by credit scheduler. To guarantee internal non-blocking in switching fabric, per flow should be allotted to all switch cells in central-stages averagely.

Suppose a certain output port and an ideal traffic model, we follow the filter rule when allotting the traffic from certain input port to all switch cells in central-stage. And we also suppose: in a slot, there are k ports in central-stage is free, R is the bandwidth of the whole fabric, ΔB is the bandwidth allocation parameter of certain flow, A_x stands for the x th switch cell in input-stage, while B_y stands for the y th switch cell in central-stage, the credit scheduler in input-stage calculate the weighted credit that it received. If in a slot, the result satisfies the rule I:

$$l(A_x \to B_y, T) = \max\{w(A_x B_y)\} \tag{I}$$

At the same time, the result satisfies the rule II:

$$\sum l(A_x \to B_y, T) \le \frac{R}{k} + \Delta B \tag{II}$$

Then the scheduler will acknowledge the corresponding grant.

In other words, the allotment of grants and load must make sure that all flows on $A_x \to B_y$ are restricted in $R/k + \Delta B$ bandwidth. The grants in central-stage are generated independently. Easy to say, if only there are free buffers, the link between $A_x \to B_y$ will never be free. So in this ideal traffic model, all flows to enter the fabric will find free buffer. Figure 4 gives a simple example.

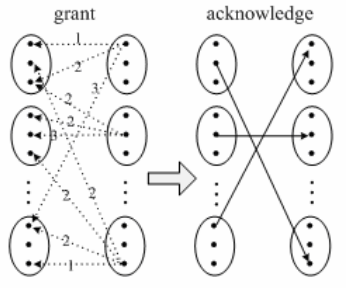

Fig. 4. A simple example of load balance

In ideal stream model, it's easy to achieve perfect load balance by rule I, but in practical system, several inputs of the same switch cell may send concurrently to the same switch cell, which is inevitable under distributed and independent load-balancing. So in order to deal with quantization imbalance, all flows between the $A_x \to B_y$ must satisfy rule II, and as a result, buffers must be set at the input-stage to store the temporary congested flows.

There are many advantages taking algorithm for load-balance in central-stage: the algorithm is brief, expansible and prone to be implemented by hardware; the algorithm guarantees the fairness by setting the upper limit of bandwidth, while the credits are generated based on certain QoS strategy, so it can efficiently support quality of service also; the fabric achieves perfect load-balancing for per packet and can control the congested flow in real-time.

3.4 Analysis of the Scheduling Algorithm

Based on the analysis above, we put forward an iSLIP alike scheduling algorithm based on the CB-3Clos switching fabric—RGA algorithm. It takes advantages of parallel iterative matching algorithm—PIM and round-robin matching algorithm—iSLIP, adopts the PIM's "request-grant-accept" mechanism and iSLIP's "Round-Robin" mechanism, is distributed implemented in credit schedulers, so this algorithm can achieve fairness and stabilization while keeps high efficiency.

The RGA algorithm still has three steps in sum: they are request, grant and acknowledgement.

Step 1: request. All input ports whose VOQ is busy send requests to the corresponding output ports.
Step 2: grant. Once receive several requests from input ports, the credit schedulers in output-stage generate credits based on the state of buffers and queues under certain QoS strategy. After inform the credits to the input ports, the credit schedulers increase the poll point by one (mol N).
Step 3: acknowledgement. After receive the grants, the unmatched input ports select the output ports satisfying the rule I and II from the credit queues to acknowledge. If it's the first time, the input ports and the acknowledged output ports should modify the poll point. The modification rule is: the points of both input and output port increase by one (mol N). After each match, all unmatched ports turn into the next match.

4 Analysis of Simulation

To illuminate the performance of RGA algorithm, we establish the simulation model. Under this model we get the throughout and packet delay and then compare them with the performance of CRRD algorithm in MSM fabric, All through the simulation supposing $m = 2n$.

4.1 Traffic Model

First we define the traffic model in the simulation model.

- Uniform traffic: the ports aimed by arriving flows distribute uniformly, and all ports face the same traffic load.
- Unbalanced traffic: the load ρ between input port s and output port d follows:

$$\rho_{s,d} = \begin{cases} \rho\left(\omega + \dfrac{1-\omega}{N}\right), & if \quad s = d \\ \rho\dfrac{1-\omega}{N}, & otherwise \end{cases} \tag{1}$$

ω stands for the unbalanced factor, N stands for the number of ports, so the load on d is:

$$\rho_d = \sum_s \rho_{s,d} = \rho\left(\omega + N\frac{1-\omega}{N}\right) = \rho \tag{2}$$

In the model, we take the Bernoulli traffic and burst traffic, In the same period of ON, the destination addresses are same, while in different period, the destination addresses are distributed in uniform and unbalanced model, the length of both ON and OFF state follow the geometry distribution, by the traffic load ρ and average burst length E_{om}, we will get:

$$\rho_d = \frac{1}{E_{ON}}, q = \frac{\rho \times p}{1 - \rho + \rho \times p} \tag{3}$$

4.2 Performance Comparison Between Algorithms with Fix-Size Packets Under Different Traffic Models

4.2.1 Uniform Bernoulli Traffic Model

Table 1 shows the throughout of RGA and CRRD algorithm under fabrics of different scales by uniform Bernoulli traffic model, it's easy to say that both algorithms achieve high throughout, especially RGA algorithm nearly get 100% throughout.

Table 1. Throughout of RGA and CRRD algorithm under fabrics of different scales by uniform Bernoulli traffic model

	64×64	128×128	256×256
CRRD	96.4%	97.1%	95.2%
PGA	99.3%	99.1%	98.9%

4.2.2 Unbalanced Traffic Model

Under unbalanced traffic model, the performance of RGA and CRRD algorithm shift dramatically.

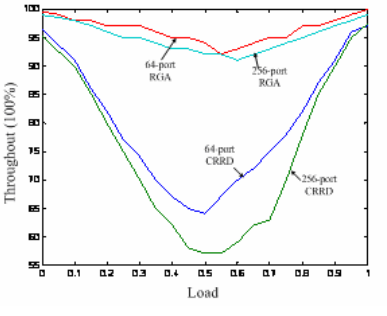

Fig. 5. Throughout of algorithms under fabrics of different scales by unbalanced traffic model

As figure 5 shows, under fabrics of different scales by unbalanced traffic model, throughout of both algorithms drop fast, and achieve the minimum 57% when ω is 0.5 and then ascend. But noticeably, the throughout of RGA algorithm holds above 90% all the time, which illuminates that the throughout of RGA algorithm is irrespective with arriving traffic model.

Figure 6 gives the packet delay of algorithms under fabrics of different scales by unbalanced traffic model. It's obvious that the packet delay increases with ω fast.

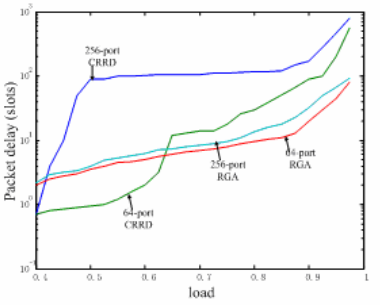

Fig. 6. Delay of algorithms under fabrics of different scales by unbalanced traffic model

Under 64-ports, when ω is under 0.65, the packet delay of RGA is a little bigger than CRRD's, but when ω exceed 0.65, the packet delay of RGA is smaller than CRRD's a lot. It's the same under 256-ports. This phenomenon can be explained by the buffers set in the central stage of switching fabric. To achieve better load balance, the buffers will work out certain delay, but while ω exceeds certain ambit, the impact of buffers reduce.

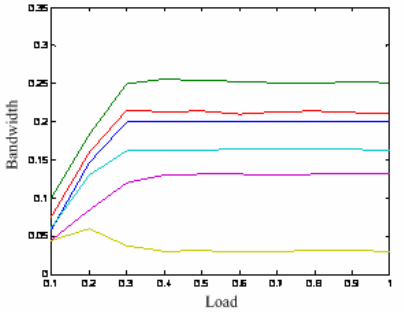

Fig. 7. The bandwidth allocation of RGA

Figure 7 gives the bandwidth allocation of RGA. Supposing the arriving rate of EF and AF is: 18%, 24%, 20%, 16%, 12%, 10%. The corresponding bandwidth allocation is: 19.8%, 24%, 20%, 16%, 12%, 8.2%. From the figure 7 we can see: the performance on bandwidth allocation of RGA is good.

5 Conclusions and Future Work

On the basis of analysis of congestion control theory, we put forward a new central-stage buffered three-stage Clos switching fabric—CB-3Clos and the backpressure-based flow control strategy under credit-dispensed mode. By analyzing the condition to satisfy the central-stage load balance, we also advance an iSLIP alike scheduling algorithm—RGA. The simulation results show: compared with CRRD algorithm of

MSM Clos fabric, the RGA algorithm has high throughput irrespective with the arriving traffic model and better performance in packet delay. At the same time, the QoS can be guaranteed.

By setting buffers at the central-stage, we can reduce the complex of algorithm while improve the performance of switching fabrics, and current circuit level make it possible. But this fabric with multi-route may lead packets to out-of-order, how to keep the order of packets in multi-stage fabrics will be our emphases of researches in the future.

Acknowledgment

This paper is jointly funded by Chinese National High Technology Research and Development Program (NO. 2005AA121210) and the National Science Foundation of China (NO. 60572042)

References

1. Clos, C.: A study of nonblocking switching fabric networks [J]. BSTJ 32(5), 406–424 (1953)
2. Chao, H.J., Deng, K., Jing., Z., Petabit, A.: Photonic Packet Switch (P3S) [C]. In: Proc. IEEE Infocom 2003, vol. 21(7), pp. 1096–1112. IEEE Computer Society Press, Los Alamitos (2003)
3. Oki, E., Jing, Z., Rojas-Cessa, R., et al.: Concurrent Round-Robin-Based Dispatching Schemes for Clos-Network Switches [J]. IEEE/ACM Trans. on Networking 10(2), 830–844 (2002)
4. Sapountzis, G., Katevenis, M.: Benes Switching Fabrics with O(N)-Complexity Internal Backpressure [J]. IEEE Communications Magazine 43(1), 88–94 (2005)
5. Kong, H.T., Morris, R.: Credit-Based Flow Control for ATM [J]. IEEE Magazine 9(2), 40–48 (1995)
6. Duato, J., Johnson, I., Flich, J., et al.: A New Scalable and Cost-Effective Congestion Management Strategy for Lossless Multistage Interconnection Networks [C]. In: Proc. HPCA-11, San Francisco, USA, pp. 108–119 (February 2005)
7. Chrysos, N., Katevenis, M.: Scheduling in Switches with Small Internal Buffers [C]. In: Proc. IEEE Globecom2005, MO, USA, pp. 614–619 (2005)
8. Chang, C., Chen, W., Juang, H.: On Service Guarantees for Input Buffered Crossbar Switches: A Capacity Decomposition Approach by Birkhoff and von Neumann [C]. In: Proceedings of IEEE IWQoS, pp. 79–86 (1999)
9. Chao, H.J., Park, J.S.: Centralized contention resolution schemes for a large-capacity optical ATM switch [C]. In: Proc. IEEE ATM workshop, pp. 11–16 (1998)