Fair Service Analysis of Load Balanced Birkhoff von Neumann Switches

Srđan Durković and Zoran Čiča

Abstract — **Modern packet networks must support high speed packet transmissions. Routers and switches have a major impact on the packet network performance. Several high performance packet switch architectures have been proposed in the literature. One of the most popular packet switch solutions are load-balanced Birkhoff von Neumann switches (LB-BvN) which comprise two switching stages. LB-BvN switches are very popular because the configurations of the switching stages are deterministic, hence it is not necessary to recalculate switch configurations in real time. This property enables high scalability of the LB-BvN switch architecture. Very important feature of packet switches is fair service when some of the output ports are overloaded. The switch should provide fair share of overloaded port's capacity to all flows destined to the overloaded output port. This paper analyzes the fair service of the most popular LB-BvN based switches.**

Index Terms— **packet switching, Birkhoff von Neumann architecture, fair service.**

I. Introduction

Implementation of high speed optical networks enabled higher transmission rates and larger traffic volumes in the core network. However, it is necessary that network equipment (routers, switches) is able to support these increased transmission speeds and larger traffic volume. Therefore, a lot of attention is given to enhancement of the router performances, especially to packet switches inside the routers. Very popular packet switch solution is LB-BvN architecture. LB-BvN comprises two switching stages. The switch configurations in both stages are deterministic. The role of the first stage is to balance traffic, while the second stage forwards packets to appropriate output ports. The goal of the first stage is to make the traffic as uniform as possible and hence enable the usage of deterministic configurations in both switching stages. This means that there is no need to recalculate switch configurations in the real time, i.e. connection patterns between input and output ports, because the switch configurations are known in advance and periodically repeated. This feature makes LB-BvN significantly simpler for implementation than other types of switches. The main problem of LB-BvN is 'packets out-of-order' problem which is the consequence of the traffic balancing. Several solutions have been proposed to solve the 'packets out-of-order' problem.

LB-BvN switches can be grouped in three categories: switches that use resequencing buffers at the output ports, frame based switches, and switches that use feedback mechanism for control communication. The popular solutions that belong to the first group (LB-BvN that use the resequencing buffers) are First Come First Served (FCFS) [1], Earliest Deadline First (EDF) [1], Byte-Focal (BF) [2]. In these solutions, the resequencing buffers at the output ports are used to reorder packets. The main goal is to bound the resequencing buffer size, typically to O(N^2). The drawback of these solutions is that the usage of the resequencing buffers increases the overall hardware complexity of the switch.

Full Ordered Frames First (FOFF) [3], Padded Frames (PF) [4], Contention and Reservation (CR) [5] switches are popular solutions that use the frame based approach. The packets are not sent to the output ports separately, but in frames. Frame represents the group of N packets that belong to the same flow, where N is the number of input/output ports. Flow is defined as a set of all packets that arrive to the same input port i and are destined to the same output port j. Transmission of frames instead of individual packets guarantees the same delay for all packets in the frame, hence there are not out of order packets in the flow. However, these routers have significantly lower performances in terms of average packet delay than the other LB-BvN solutions. Packet delay under lighter loads is much higher because a large amount of time is needed to complete the frame, which induces high packet delays at the input ports.

The third group of LB-BvN switches represent the switches that use feedback mechanism for control communication [6-8]. These solutions use specific connection patterns between input and output ports in order to provide good feedback path for efficient control communication. Feedback mechanism enables the input ports to adequately choose packets for sending to avoid the packet out-of-order problem. The most popular solutions in this third group of LB-BvN switches are mailbox switch [6], and feedback switch with staggered symmetry (FS) [7-8].

Papers that proposed LB-BvN based solutions analyzed the performance of the proposed solutions in terms of switch throughput, average and maximum packet delay. But, fair service, which is also very important switch property, was not analyzed. In this paper, fair service of the most popular LB-BvN solutions is analyzed. Fair service is very important feature that can significantly affect quality of service for end users as well as the overall network performance. Fair service can also minimize the effect of the malicious attacks that could take advantage of fair service absence where attacker's aggressive flows can consume almost complete capacity of the overloaded output port. In this paper, BF, CR and FS switches are analyzed, because they are the best representatives of the switches that use resequencing buffers, the frame-based switches and the switches that use feedback mechanism, respectively.

The remainder of the paper is organized as follows. In the next section, we give a short description of the analyzed

Srđan Durković and Zoran Čiča are with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (e-mail: srdjad6@gmail.com, zoran.cica@etf.bg.ac.rs).

solutions: BF, CR and FS. In the third section, we give a description of the fair service tests that were used in this paper to inspect the fair service in tested LB-BvN solutions. In the fourth section we present the results of the fair service tests conducted on selected LB-BvN switches, while the last section concludes the paper.

II. Analyzed LB-BvN solutions

In this paper, three LB-BvN based switches are analyzed: BF [2], CR [5] and FS [7-8]. BF switch has the best performance in terms of throughput and average packet delay among the LB-BvN switches that use the resequencing buffers. CR is chosen because it achieves better performance than the other frame-based switches, although CR switch is actually a combination of frame-based approach (under heavier loads) and feedback mechanism approach (under lighter loads). Finally, FS is chosen because it achieves the best performance among all LB-BvN switches in terms of average packet delay. In this section we give a short description of all three selected LB-BvN switch solutions.

BF comprises two switching stages like all other LB-BvN switches. Central ports are introduced between these two stages. Deterministic connection patterns between input and central ports (the first switching stage), and central and output ports (the second switching stage) are as follows: the input port i, in the time slot t is connected to the central port j, where $j=(i+t)\%N$, while the central port j in the time slot t is connected to the output port k, where $k=(j+t)\%N$. At the input ports, packets are stored in the queues according to flows they belong to. For each flow there is a pointer that points to which central port the next packet of that flow should be sent. When the input port i is connected to the central port j, the input port i selects the packet from the queue whose pointer points to central port j. If there are several such candidate queues, then the input port i selects the packet from the longest candidate queue. At the central ports, packets are stored in the queues according to their final destination output port. When central port j is connected to the output port k then central port j sends the packet from the queue that stores the packets for output port k. Each output port implements resequencing buffers to reorder out of order packets.

CR switch represents combination of frame-based switches and switches with feedback mechanism. CR switch also comprises two deterministic switching stages. Input and output ports i and j are connected according to patterns: $(i+j)\%N=(t+1)\%N$. Each input port implements N VOQ queues, while at each central port there are N I-VOQ queues (VOQ with insertion). CR works in two modes: contention and reservation mode. Frame represents group of N packets that belong to the same flow, while time frame represents the sequence of N consecutive time slots. For the input port i, time frame begins at the time slot when the input port i is connected to the first central port. The input port selects its working mode at the beginning of its time frame. If there are more than $N\text{-}1$ packets in some VOQ at the beginning of time frame, then the input port works in the reservation mode, otherwise the input port works in the contention mode. If there are multiple VOQs with more than $N\text{-}1$ packets, the round-robin principle is used for selection of the VOQ. In the reservation mode, the input port sends packets from the chosen VOQ to the central ports in the next N time slots. The sent packets are stored to the end of the I-VOQs at the central ports. In the case of contention mode, the non-empty VOQ is selected according to round-robin principle. Packets from the selected VOQ are sent to the central ports. If in the appropriate I-VOQ there is a fake packet, then that fake packet is deleted and the packet received from the input port (that works in contention mode) is stored. Fake packet is a dummy packet which is stored in the I-VOQ when the I-VOQ gets empty. If there is no fake packet in I-VOQ, then the packet is rejected, and the information about the rejection is sent to the input port using the feedback mechanism. Under heavier loads, less time is needed for completing the frames, thus, input ports dominantly work in reservation mode (like in frame-based switches), while under lighter loads input ports dominantly work in contention mode.

FS achieves the best performance in terms of average packet delay under various admissible traffic scenarios among other LB-BvN switches [7-8]. FS also comprises two deterministic switching stages. At the input ports there are N VOQs, where each queue corresponds to one flow. However, at the central ports only one packet can be stored for each output port. Two switching stages have deterministic configurations, but in a different way compared to other LB-BvN switches. The connections between the input ports and the central ports, and between the central ports and the output ports have so called staggered symmetry and in-order packet delivery features. Staggered symmetry feature means that if the central port j is connected to the output port k in the time slot t, then in the time slot t+1 the input port k will be connected to the central port j, which enables efficient exchange of control information. At each central port there is occupancy vector which shows which VOQs are occupied. Thanks to staggered symmetry feature this vector is forwarded from the central port j to the output port k, and consequently to the input port k because input and output port are implemented on the same line card. Since input port k is connected to the central port j in the next time slot, then the input port k knows from which flows a packet can be sent, without violating the rule that only one packet can be stored for each output port at central port j. In-order-packet-delivery feature guarantees that all packets from the same flow have the same delay through the switch. In this way, packet out-of-order problem is avoided.

III. Fair service tests

The LB-BvN switches have been analyzed in terms of throughput, average and maximum packet delay for various admissible traffic scenarios. However, there have been no analysis in terms of fair service. Fair service is very important feature, which shows how switch serves affected flows that share the capacity of the overloaded port. In admissible traffic conditions when none of output ports is overloaded all three analyzed LB-BvN switches (BF, CR, FS) achieve 100% throughput. However, it is interesting to see how these switches behave under non-admissible traffic scenarios, when one output port is overloaded. If there is no fair service, then it is possible that one flow occupies most of the overloaded output port's capacity and thus disable servicing of the other flows that are destined for that overloaded output port. Fair service implies that all flows should evenly share the capacity of the overloaded output port. To evaluate fair service, we analyze three traffic scenarios in this paper: 1) all input ports send packets to the same (overloaded) output port; 2) *X* input ports send packets to the same (overloaded) output port, and other input ports send packets uniformly to all output ports; 3) *X* input ports send packets to the same (overloaded) output port, while other input ports send packets in hot-spot manner [2]. It is assumed that switches have *N* input and *N* output ports, and input load is 100% at each input port, which means that at each input port one packet arrives at each time slot. It is assumed that all flows have the same weight.

In the scenario 1, all input ports send packets to the same output port. Fair service in this case means that each flow should get *1/N* of the overloaded output port's capacity. If *N*=32, then each flow should get 3.25% of the overloaded output port's capacity.

In the scenario 2, *X* input ports send packets only to the overloaded output port, while other input ports uniformly send packets to all output ports. If the output port 1 is the overloaded output port, *X* input ports (that send packets only to output port 1) will have *Y* packets for the output port 1, while other input ports will have about *Y/N* packets to output port 1, where *Y* is the number of the observed time slots. Fair service in this case means that each flow gets *1/N* of the capacity of the overloaded output port. If *N*=32, then each flow should get 3.25% of the overloaded output port's capacity. Other output ports are not overloaded and they get about *Y/N* packets from each of *N-X* input ports (which generate uniform traffic). The expected share in the capacity of the output ports for the scenario 2 is summarized in table 1, when *N* is set to 32 (the values given in table 1 do not depend on value of *X*).

TABLE 1. EXPECTED DISTRIBUTION OF CAPACITY SHARE AT THE OUTPUT PORTS IN THE SCENARIO 2

Outputs / Inputs	Overloaded output	Other outputs
X	3.125%	0%
N-X	3.125%	3.125%

In the scenario 3, *X* input ports send packets only to the overloaded output port. Other output ports send packet according to hot-spot scenario [2]: input port *i* sends 50% packets to the output port *i*, while to the other output ports it sends packets uniformly, i.e. about 50%/(*N*-1) packets. In the scenario 3 there are two subcases. Let us assume that the output port 1 is overloaded. In the subcase 1, the input port 1 is one of the *X* input ports that send packets only to the output port 1, and in the subcase 2, the input port 1 isn't one of those *X* input ports. In fair service terms, at the overloaded output port 1 each flow should get *1/N* of the capacity of the output port. However, flows from *N-X* input ports that send packets according to hot-spot scenario where overloaded output port isn't their hot-spot output port require only 1/2(*N*-1) of the capacity of the output port 1. Thus, their non-used capacity is equally shared between flows that originate from the *X* input ports that send packets only to output port 1 (subcase 1), or between flows that originate from the *X* input ports that send packets to output port 1 and the flow from the input port 1 (subcase 2). Tables 2 and 3 summarize the expected share in the capacity of the output ports according to fair service for scenario 3, when *N* is set to 32 and *X* is set to 8. In Table 3, the row *I* represents the input port whose hot spot output is the overloaded port, and that input port is not in the set of *X* input ports that send packets only to the overloaded port.

TABLE 2. EXPECTED DISTRIBUTION OF CAPACITY SHARE AT THE OUTPUT PORTS IN THE SCENARIO 3 (SUBCASE 1)

Outputs / Inputs	Overloaded output	Hot spot output	Other outputs
X	7.7%	0%	0%
N-X	1.6%	50%	1.6%

TABLE 3. EXPECTED DISTRIBUTION OF CAPACITY SHARE AT THE OUTPUT PORTS IN THE SCENARIO 3 (SUBCASE 2)

Outputs / Inputs	Overloaded output	Hot spot output	Other outputs
X	7%	0%	0%
I	7%	-	1.6%
Others	1.6%	50%	1.6%

IV. SIMULATION RESULTS

In this section we show the fair service analysis results for the inspected LB-BvN solutions (BF, CR, FS). We assume that switches have 32 input ports and 32 output ports (*N*=32). The number of input ports that send packets only to the overloaded output port is set to 8 (*X*=8). To evaluate fair service, we observe the flows at the overloaded output port. We compare the number of the packets that were sent from the overloaded output port for each flow with the expected number of packets calculated according to values given in Tables 1-3 and the number of time slots in the simulation. We show the simulation results in Figures 1-4. where *x*-axis represents the id number of the input ports, and *y*-axis represents the number of packets that were sent from the output port for the corresponding flows. Dashed line represents the expected number of packets in the case of the ideal fair service calculated according to values in Tables 1-3 and the number of time slots in the simulation. We have also performed simulation for other values of *N* and *X*, and those results confirm the results given in figures 1-4. Thus, we do not show these results to avoid redundancy. Additionally, for all three scenarios, our simulations showed that the selection of the input ports which send packets only to the overloaded output port is not important, i.e. any selection will give the same results. The same conclusion applies for the selection of the overloaded output port. Therefore, the first *X* ports are selected to be the ports that send packets only to the overloaded output port. The overloaded output port in results shown in Figures 1-3 is output port 1. For the scenario 3 subcase 2 shown in Figure 4, the overloaded output port is output port 32 (the reason is to avoid a change of the selected *X* ports in previous scenarios). Note that in all simulated scenarios non-overloaded output ports work properly and no traffic is lost on these output ports.

Figure 1 shows the results for the scenario 1. BF and CR achieve ideal fair service in the scenario 1, but FS has very poor performance in terms of fair service. In the case of FS, the first input port of the input ports from the set of *X* ports that starts to send packets to the overloaded output port will prevent all the other input ports to send packets to the overloaded output port. This is consequence of the fact that at each central port there can be only one packet for each of the output ports. Once, one of the input ports from the set of *X* ports starts sending packets it will always consume that one free space at each of the central ports, thus blocking all the other flows destined to the overloaded output port.

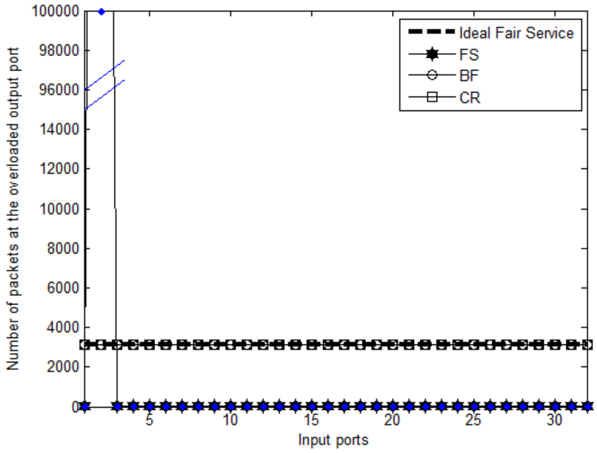

Figure 1. Distribution of packets at the overloaded output port in scenario 1

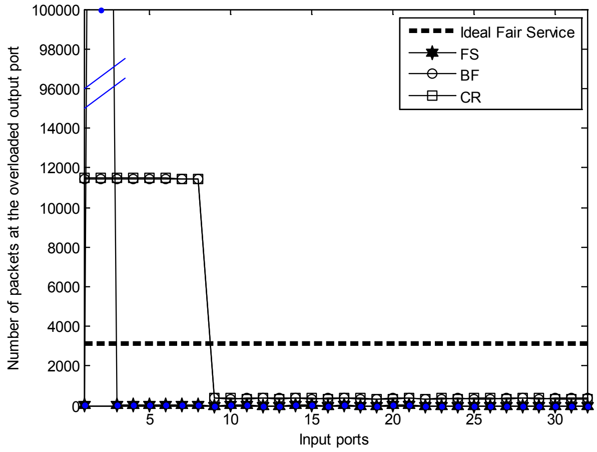

Figure 2. Distribution of packets at the overloaded output port in scenario 2

Figure 2 shows the results for the scenario 2, when *X*=8 input ports send packets only to the overloaded output port, while others send packets uniformly to all output ports. Again, FS has the worst performance in terms of fair service for the same reasons as in scenario 1. BF and CR still achieve better results, but now their performance significantly differs from the ideal fair service performance, as shown in figure 2. In CR switch, input ports that send packets only to the overloaded output port create frames faster and work in reservation mode. Other input ports dominantly work in contention mode, but as there are no fake packets at central ports (*X* input ports will always have frames for sending which prevents appearance of fake packets) those input port will have to wait to create their frames. Hence, aggressive flows take larger share of the links capacities between the central and output ports, and consequently take a larger share of the overloaded output port's capacity. In the case of BF, input ports which send packets only to the overloaded output port synchronize with deterministic switch configuration at one moment and after that moment those input ports start to send packets to the central ports constantly. Similarly to CR, aggressive flows take a larger share of the links capacities between the central and output ports, hence they take a larger share of the overloaded output port's capacity.

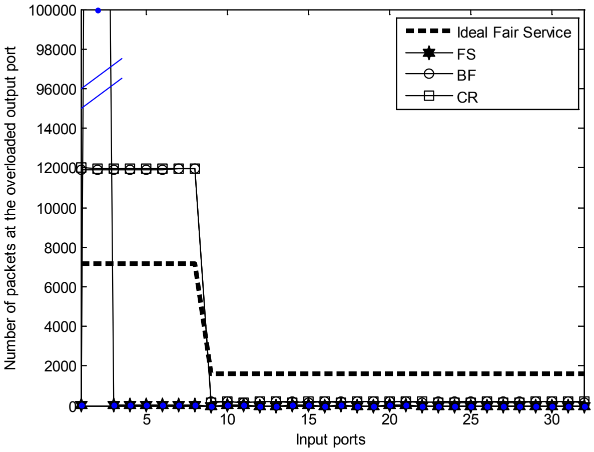

Figure 3. Distribution of packets at the overloaded output port in scenario 3 subcase 1

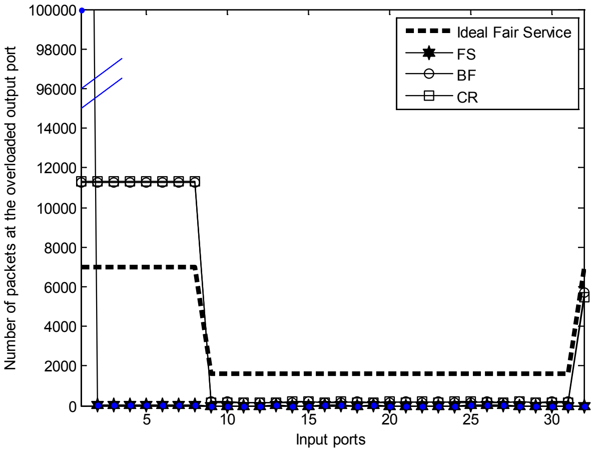

Figure 4. Distribution of packets at the overloaded output port in scenario 3 subcase 2

Figures 3 and 4 show the results for the scenario 3, both subcases. The results are very similar to the results shown in Figure 2. FS again has the worst performance, while BF and CR significantly differ from the ideal fair service performance. The reasons for the poor fair service performance are the same as explained earlier in this section.

V. Conclusion

In this paper, we analyze the most popular LB-BvN switch solutions (BF, CR, FS) in terms of fair service. Analysis results show that existing LB-BvN solutions are very unfair. FS is extremely unfair, because one aggressive flow gets all the resources on the overloaded output port. BF and CR achieve better performance than FS, but aggressive flows also get higher share in the capacity of the overloaded output port when compared to non-aggressive flows. Thus, a further research should be taken in order to improve the LB-BvN switch performance in terms of fair service.

REFERENCES

[1] C. S. Chang, D. S. Lee, and Y. S. Jou, ”Load balanced Birkhoff-von Neumann switches, part II: multistage buffering,” *Computer Communications*, vol.25, no.6, pp.623-634, 2002.

[2] Y. Shen, S.S. Panwar, H.J. Chao, ”Design and performance analysis of a practical load-balanced switch,” *IEEE Transactions on Communications*, vol.57, no.8, pp.2420-2429, 2009.

[3] I. Keslassy, and al., ”Scaling internet routers using optics,” *Proc. of SIGCOMM 2003*, Karlsruhe, Germany, Aug. 2003.

[4] J.J. Jaramillo, F. Milan, and R. Srikant, “Padded frames: a novel algorithm for stable scheduling in load-balanced switches,” *IEEE/ACM Transactions on Networking*, vol.16, no.5, pp.1212-1225, 2009.

[5] Chao-Lin Yu, Cheng-Shang Chang, Duan-Shin Lee, “CR switch: A load-balanced switch with contention and reservation,” *IEEE/ACM Transactions on Networking*, vol.17, no.5, pp.1659-1671, 2009.

[6] C.S. Chang, D.S. Lee, Y.J. Shih, C.L. Yu, “Mailbox switch: a scalable two-stage switch architecture for conflict resolution of ordered packets,” *IEEE Transactions on Communications*, vol.56, no.1, pp.136-149, 2008.

[7] B. Hu, K. Yeung, “Feedback-based Scheduling for Load-balanced Two-stage Switches,” *IEEE/ACM Transactions on Networking*, vol.18, no.4, pp.1077-1090, 2010.

[8] A. Huang, B. Hu, “The Optimal Joint Sequence Design in the Feedback-based Two-stage Switch,” *Proc. of ICC 2014*, Aug. 2014.