

Packet Delay Variation Analysis of the CQ Switch

Milutin Radonjić, *Member, IEEE*, Igor Radusinović, *Member, IEEE*, and Ivo Maljević, *Member, IEEE*

Abstract — Packet delay variation analysis for a 32x32 crosspoint queued switch under uniform and IBP arrival traffic is presented in this paper. Different buffer sizes and several scheduling algorithms are observed: longest queue first, round robin, exhaustive round robin and frame based round robin matching. Results show that the least packet delay variation, among observed algorithms, causes round robin algorithm.

Keywords — Buffer length, crossbar switch, delay variation, scheduling algorithm.

I. INTRODUCTION

THE crossbar switch architecture is one of the most popular architectures for building modern high-speed switches and routers. The crossbar switching fabric is a favorable choice of implementation because of its simplicity and internal non-blocking capability. Most of the crossbars considered in the literature, as well as the most widely used crossbars in commercial products, are unbuffered crossbars. In these cases, buffering has to be done outside of a switch fabric, on inputs, outputs, or both.

The possibility of buffering packets inside the switching fabric was not seriously considered for a long time. The technology limitations made this approach impractical for implementation. Consequently, there was no interest in deeper analysis and evaluation. Switches with small buffers in crosspoint (usually one-cell length) that work in accordance with larger buffers at inputs or/and outputs have been evaluated [1].

However, it has been recently shown that larger buffers can be implemented on the same chip with switching fabric, using modern technology [2]. The switch with buffers only in crosspoints of switching fabrics is called CQ (Crosspoint Queued) switch. Absence of input buffers is a major advantage of this architecture. Linecards, where input buffers reside, require control communication with a centralized scheduler in order to notify their occupancies. This control communication is time-consuming and therefore overall switch performance is limited.

This work is supported in part by the Montenegrin Ministry of Science under grant 01-451/2012 (FIRMONT).

Milutin Radonjić is with the School of Electrical Engineering, University of Montenegro, Bulevar Džordža Vašingtona bb, Podgorica, Montenegro (phone: 382-20-245839; e-mail: m.radonjic@iee.org).

Igor Radusinović is with the School of Electrical Engineering, University of Montenegro, Bulevar Džordža Vašingtona bb, Podgorica, Montenegro (phone: 382-20-245839; e-mail: igorr@iee.org).

Ivo Maljević is with the TELUS Mobility, 200 Consilium Place, Suite 1300 Scarborough, Ontario, Canada, M1H 3J3; (e-mail: ivom@iee.org).

The following parameters are widely used for crossbar switch evaluation: throughput, average cell latency, and cell loss probability. It is a common assumption that incoming packets have a fixed size length (referred to as cells), which means that segmentation and reassembly are accomplished outside the switch (at inputs and outputs) [3].

In our previous papers we deeply analyzed CQ switch performance with several scheduling algorithms, different buffer lengths, and various incoming traffic models [4]-[7]. These analyses showed the way that buffer length and scheduling algorithm affect throughput, cell loss probability and average latency of cells traversing the switch.

However, average cell latency is not the only parameter that describes delay of cells during their transfer through the switch. Namely, it is very important to identify and observe differences in the delays of individual cells, which are part of the same packet. The packet cannot be reassembled until all cells belonging to that packet arrive to the output port. Hence, if only one cell from the same packet has a long delay, and others pass through the switch with low latency, the entire packet will be delayed at output as a cell with the largest delay. This means that an effective packet delay is much larger than the average cell latency, because the calculation of average latency considers cells that had no delay. On the other hand, if all the cells from the same packet have an equal delay, then the effect of this delay is canceled at the destination by taking the moment of arrival of the first cell of that packet as the initial moment.

From these examples it can be concluded that large differences in delay between cells significantly degrade the effective switch performance. A parameter that describes this effect is *IPDV* (Instantaneous Packet Delay Variation) [8].

IPDV is defined as the difference in one way delay between successive packets, ignoring lost packets. Given a sequence of packets transmitted at times $t(1), t(2), t(3), \dots, t(n)$ and received at times $t'(1), t'(2), t'(3), \dots, t'(n)$, then the sequence of delays is $d(1), d(2), d(3), \dots, d(n)$, where $d(i) = t'(i) - t(i)$ and $d(i) \geq 0$. Thus, the *IPDV*, as defined above, is the sequence $d(2) - d(1), d(3) - d(2), \dots, d(n) - d(n-1)$. Maximum *IPDV* is usually defined as: $\max \{ \text{abs}[d(2) - d(1)], \text{abs}[d(3) - d(2)], \dots, \text{abs}[d(n) - d(n-1)] \}$.

It is also important to observe how many cells have certain *IPDV* values. For that reason, we will present the distribution diagrams of cells over *IPDV*, where

distributions are calculated as the number of cells with a particular *IPDV* value divided by the total number of transferred cells.

In this paper, *IPDV* analysis is performed for various scheduling algorithms and different buffer lengths under the uniform and Interrupted Bernoulli Process (IBP) traffic conditions.

The rest of this paper is organized as follows: Section II gives a description of observed scheduling algorithms. In Section III we present our simulation model. Section IV describes the simulation results under the uniform traffic. Simulation results under the IBP incoming traffic are given in Section V. We draw conclusions in Section VI.

II. ANALYZED SCHEDULING ALGORITHMS

The choice of the buffer whose head-of-line cell will be forwarded to the appropriate output can be performed using any of the well-known scheduling algorithms. In this paper the following scheduling algorithms are observed: Longest Queue First (LQF) [9], Round Robin (RR) [10], Exhaustive Round Robin (ERR) [11] and Frame-based Round Robin Matching (FRRM) [12].

LQF algorithm services buffer with the highest occupancy on a particular output line, in each time slot.

RR algorithm services occupied buffers on a particular output line in the circular (round robin) order, handling all buffers without priority. After the departure of one cell from the buffer, it will service the next occupied buffer in the following time slot. The service is always performed in the same order determined by the buffer position.

ERR algorithm also services buffers on the same output line in the round robin order. However, this algorithm continuously services the same buffer until it becomes empty and after that passes to the next occupied buffer.

FRRM algorithm services the same buffer until it departs all the cells that were in the buffer at the time it started to be serviced. The buffer occupancy at the time it starts to be serviced is called a “frame.” After departure of all cells from the frame, the next buffer on the same output line is serviced in the round robin order.

ERR and FRRM algorithms are based on RR algorithm and differ only in the number of cells that will be continuously forwarded from the same buffer, when it gets the right to be serviced.

III. SIMULATION MODEL

In this paper an *IPDV* at 32x32 CQ switch under the uniform and IBP traffic is analyzed. Within a uniform traffic model, each input receives one cell in each time slot with the same probability p (offered load), and this cell is intended for any output with a probability $1/N$, where N is the number of ports (inputs/outputs). Incoming cells are uniformly distributed to all output ports. It means that cell arrival probability from any input i to any output j is p/N .

Uniform traffic simulations are performed for different values of an offered load (p), ranging from 0.01 (instead of zero) to 1, with a step of 0.1. Particular attention has been paid to heavy traffic ($p \geq 0.9$), which is simulated with a step of 0.01.

We will present results for $p=0.95$. A rather large value of the input load is chosen because, due to the appearance of congestion, there come to the fore the differences in performance between scheduling algorithms.

In a bursty traffic model, the traffic at each input is modeled as the ON-OFF Interrupted Bernoulli Process [5]. At ON (busy) state the source generates cells. At OFF (idle) state, the source does not generate any traffic. Output port addresses are uniformly distributed. Each of the inputs is described by a two state ON-OFF model where both busy and idle periods are geometrically distributed. Cells of the same burst are intended for the same output (model of fragmented packet). If the input is in the ON state, it will remain in that state with probability $1-a$, and switches to an OFF state with probability a . If the input is in the OFF state, it will remain in that state with probability $1-b$, while it switches to the ON state with probability b . These probabilities are given by the following equations:

$$a = 1 / B_s, \quad (1)$$

$$b = \frac{pa}{1 + pa - p}, \quad (2)$$

where B_s is the average burst size and p is the offered input load.

When the input is about to leave the ON state, it can move to the OFF state, or can return immediately to the ON state with probability b . If it returns to the ON state, intended output can be the same as before or change to another output, according to a uniform distribution. This is the main difference between IBP and the basic ON-OFF model. Because of this modification, IBP traffic model can be simulated for a broad range of input traffic loads even for a large burst size. IBP traffic model is simulated for the same values of offered load as uniform, but with average burst sizes (B_s) of 2, 4, 8, 16, 32, and 64. In this paper we will present results for $p=1$ (maximal load) and $B_s=16$.

All simulations were performed using our own original simulator that was developed for a detailed performance analysis of CQ switches.

IV. DELAY VARIATION ANALYSIS UNDER UNIFORM TRAFFIC

Fig. 1 depicts diagrams of maximal *IPDV* values over the buffer lengths, for observed algorithms.

LQF algorithm has by far the highest maximal delay variation. For all observed algorithms, with increasing of buffer length the *IPDV* also increases, to some extent. LQF algorithm reaches the upper limit of maximal *IPDV* with buffer lengths of $L=3$, while other algorithms reach that limit with $L=16$. This means that the increase of the buffer lengths reduces the difference in the maximum delay variation between LQF and other algorithms.

Algorithms based on round robin have a fairly similar maximal delay variation. For short buffers the difference is small, while slightly more pronounced for longer buffers. Especially noticeable is the ERR algorithm, which has a higher maximal delay variation than other two, for longer buffers. RR and FRRM algorithm have values that differ in only a few time slots.

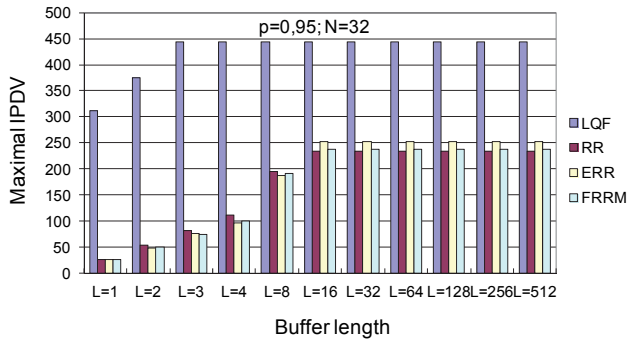


Fig. 1. Maximal values of delay variation during the simulation, for 32x32 switch under uniform traffic with offered load $p=0.95$.

The maximum value of delay variation is not a sufficient parameter for the evaluation of switch performance. Namely, it is possible that only a small number of cells have a large variation, while the dominant number of cells is transferred with the same (or similar) delay. Therefore, it is necessary to observe how many cells (in percentage) have a particular value of delay variation. Diagrams of distribution of cells for observed scheduling algorithms and various buffer lengths are shown in Fig. 2 (a - d).

The first conclusion to be drawn from these diagrams is that the distribution of cells is almost the same for the buffer length of $L \geq 3$ (for LQF), or $L \geq 16$ (for other algorithms). Reducing the buffer length below these limits reduces *IPDV* range in which the cells are distributed.

Another important feature relates to the specificity of RR algorithm. While the diagrams for other algorithms are quite symmetrical with respect to the zero *IPDV*, diagrams for the RR are extremely asymmetric. This is due to the nature of RR algorithm. It is impossible for it that a variation of delay (in accordance with the definition introduced here) to be arithmetically greater than the number of inputs/outputs of the switch. Therefore, the distribution of cells is dominantly expressed in the negative domain of the diagram.

With one-cell buffers ($L=1$) RR, ERR, and FRRM algorithm have exactly the same cell distribution over the *IPDV*. With increasing buffer lengths the difference in RR algorithm behavior occurs, while ERR and FRRM algorithms have almost the same cell distribution regardless of buffer lengths. Thus, these two algorithms have generally very similar behavior under the uniform incoming traffic.

Most cells with LQF algorithm have a low delay variance. Around 43% of cells has not any variance ($IPDV=0$) if buffers are one-cell long. This number is around 25% for longer buffers. With increasing buffer lengths, the range of delay variation values distribution also increases, but only to the length of $L=3$. Diagrams practically match for longer buffers.

There are around 18% of cells without a delay variance with RR scheduling algorithm and single-cell buffer lengths. The other cells are distributed in a narrow range of delay variations: $[-26, 25]$. With longer buffers, the amount of cells without delay variation is around 10%, but

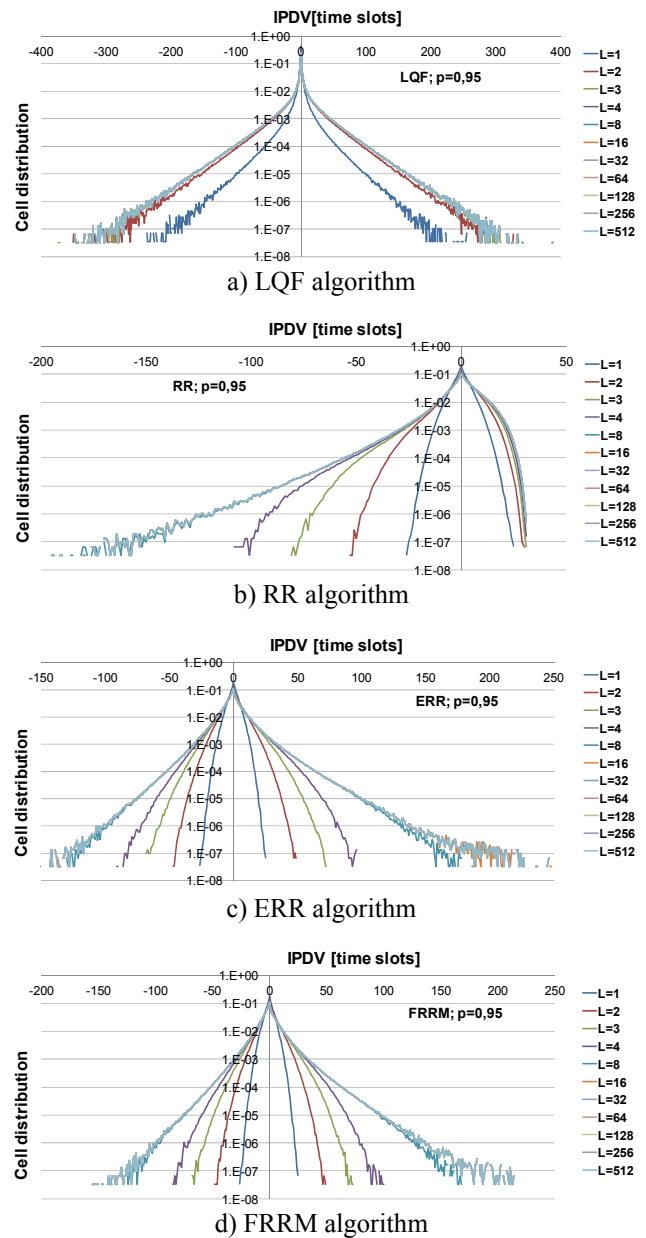


Fig. 2. Cell distribution over the *IPDV* under the uniform traffic with $p=0.95$, for different scheduling algorithms and various buffer lengths.

a range where other cells are distributed is larger (mostly in a negative domain).

ERR and FRRM algorithms have the same amount of cells without delay variation, with single-cell buffers, as RR algorithm. The amount of cells without delay variance is around 11% for a switch with longer buffers. With increasing buffer lengths, a range where other cells are distributed also increases. Unlike the RR algorithm, the positive values of delay variation are not limited by the number of switch inputs/outputs. Moreover, the extent of cell distribution range spread more dominant in the positive domain.

In order to facilitate mutual comparison of the analyzed algorithms, Fig. 3 shows their distribution of cells for a buffer length of $L=32$. The Figure clearly shows that LQF algorithm has the worst performance among the analyzed algorithms, in terms of delay variation. It has about twice

as many cells with a zero delay variation compared to the other three algorithms, but also has a significant number of cells with a large delay variation.

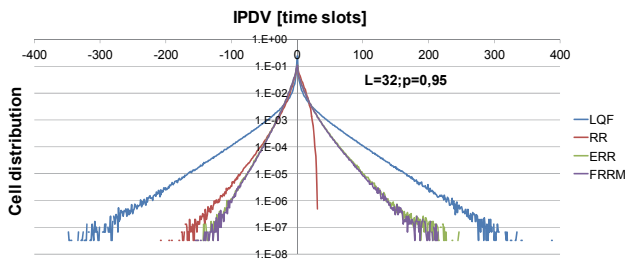


Fig. 3. Cell distribution over the *IPDV* under uniform traffic, for 32x32 switch with $L=32$ and $p=0.95$.

V. DELAY VARIATION ANALYSIS UNDER IBP TRAFFIC

IBP models large packets that are segmented into fixed-sized cells, or a large number of successive packets that are a part of the same data transfer (video or voice segments, file transfer, etc.). It is very important that such cells arrive to the destination without too much delay variation in order to be successively reassembled into original packets. Thus, the delay variation analysis is particularly important for this traffic model.

From the obtained results we can summarize some general behavioral trends. Firstly, longer buffers mostly invoke more delay variations for evaluated B_s and observed algorithms. Secondly, LQF algorithm shows significantly worse behavior compared to the other three algorithms, in terms of delay variance.

One example of obtained results is shown in Fig. 4. These diagrams present the maximal values of *IPDV*, during the simulation, for different values of buffer length with $B_s=16$ and $p=1$.

It is obvious that LQF algorithm has much higher values of maximal *IPDV* than other algorithms, especially with short buffers. Its maximal *IPDV* increases with increasing of buffer length in the range of $L \leq 32$. After that, the maximal *IPDV* will not increase regardless of the increased length of the buffers. On the other hand, the algorithms based on round robin increase maximal *IPDV* with the increasing of buffer lengths, in the whole observed range. It means that, with the increasing of buffer lengths, the difference between LQF and other three algorithms becomes smaller. For maximal observed buffer length ($L=512$) LQF algorithm has a lower value of maximal *IPDV* than other algorithms.

The algorithms based on round robin have identical performances for single-cell buffers. ERR algorithm shows much more delay variance than RR and FRRM, for buffer lengths between two and 32 cells. For $L \geq 64$ all three RR-based algorithms have comparable values of maximal *IPDV*. A general conclusion is that long buffers make the choice of algorithm less important when it comes to maximal delay variance. RR and FRRM algorithms show the best performance if buffers are short.

Further analysis refers to the observation of the amount of cells that have a particular value of delay variation. The fact that the maximal value of the delay variation of an

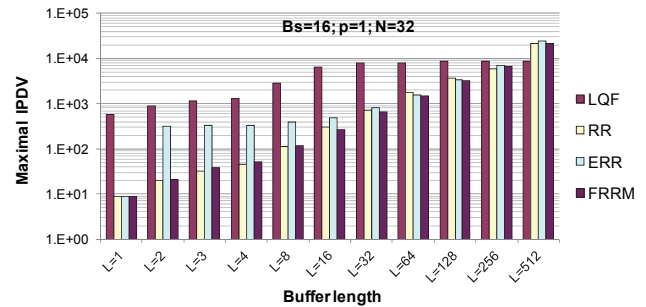


Fig. 4. Maximal *IPDV* over the buffer length for IBP traffic with $B_s=16$ and $p=1$.

algorithm is (under certain conditions) large does not necessarily mean that the algorithm is unacceptable, because it is possible that such large values are very rare. It is therefore important to know how many cells have a particular value of delay variation.

Diagrams of distribution of cells for observed scheduling algorithms and various buffer lengths, under the IBP traffic with $B_s=16$, are shown in Fig. 5 (a - d). Most cells have a small value of delay variance with LQF algorithm. This is, naturally, especially emphasized with small buffer lengths. If buffers are single-cell long, around 91% of cells have not a variance of delay ($IPDV=0$). Other cells are distributed in a relatively narrow range of values of delay variation. With increasing buffer lengths to the value of $L=32$, the amount of cells with a zero-delay-variance decreases, and the range of *IPDV* values where cells are distributed becomes wider. A further increase of the buffer lengths will not significantly change the performance. Namely, for $L \geq 64$ approximate amount of cells (around 37%) have *IPDV* equal to zero. Also, the diagrams of distribution practically overlap, i.e. the ranges where cells are distributed, as well as their number per particular *IPDV* values, are very close.

In the case of RR algorithm, delay variation is also expressed at higher buffer lengths. With single-cell buffers around 77% of cells have not a variance in delay. The remaining cells were distributed in the range $[-9,8]$. With increasing of buffer lengths, the amount of cells that have not a variation in delay decreases, even more rapidly than with LQF algorithm. For example, with buffer lengths of $L=32$ the amount of such cells is approximately 10%, while with buffer lengths of $L=512$ that amount is around 0.45%. With increasing of buffer lengths the range of cell distribution also increases, but dominantly in the negative domain of diagram. For any particular buffer length the different cell distribution can be noticed, where longer buffers invoke more delay variation.

ERR and FRRM algorithms have very similar diagrams of cell distribution over the particular *IPDV* values. With buffer lengths of $L \leq 64$ diagrams are quite symmetric against the zero variance. For longer buffers diagrams are asymmetrical because cells are mostly distributed in the positive domain of delay variations. Also, the range of delay variations where cells were distributed during the simulation is much wider in the positive domain, with longer buffers.

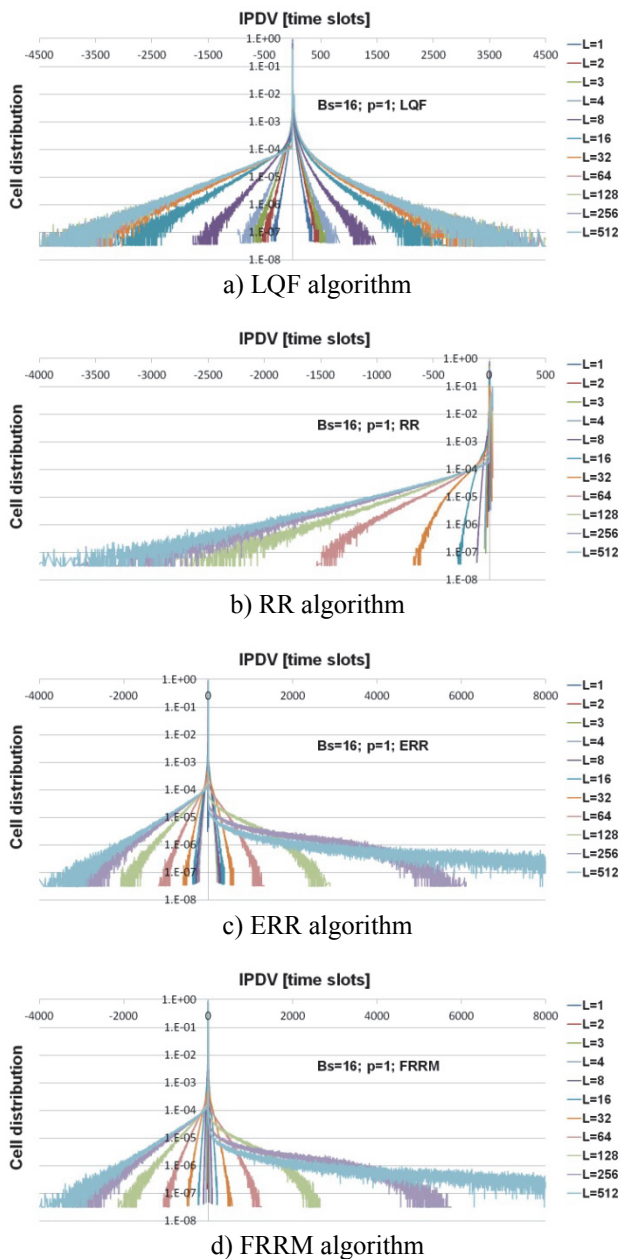


Fig. 5. Cell distribution over the *IPDV*, for different scheduling algorithms and various buffer lengths, at 32x32 switch under the IBP traffic with $p=1$ and $B_s=16$.

If buffers are single-cell long, the amount of cells without a delay variance was the same as with RR algorithm (around 77%). With increasing of buffer lengths, the amount of these cells increases, with those two algorithms. Growth is somewhat faster with ERR algorithm. For instance, with buffer lengths of $L=4$, the amount of cells that have a zero delay variance with ERR algorithm is around 89%, while this amount with FRRM algorithm is around 79%. However, with long buffers these values approach each other again. With the longest observed buffer lengths ($L=512$) the amount of cells that have a zero delay variance is around 94%, for both algorithms. Still, with large buffers the remaining cells were distributed in a wide range of delay variation, while the behavior of these two algorithms is practically the same.

The easiest way to make a clear comparative analysis of various algorithms is with the same traffic conditions and the same buffer lengths. Fig. 6 depicts results for IBP traffic with $B_s=16$ and buffer lengths of $L=128$. As can be seen from these diagrams, most of the cells have a small value of delay variation. More than 94% of cells have a zero delay variance with ERR algorithm, and almost the same amount with FRRM algorithm. With LQF algorithm that amount is around 37% and with RR algorithm it is around 1.6%.

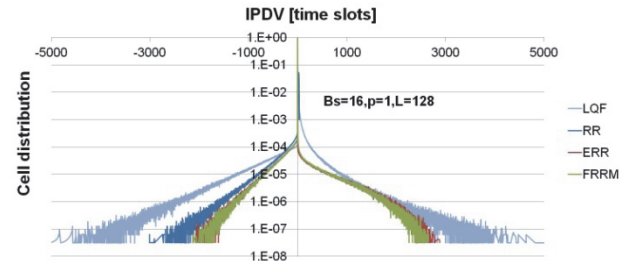


Fig. 6. Cell distribution over the *IPDV*, for 32x32 switch under the IBP traffic with $p=1$, $B_s=16$, and $L=128$.

With all algorithms, a significant amount of cells is distributed over a wide range of values of delay variation. This is the least pronounced with RR algorithm, because the positive value of the delay variation is limited by the number of ports. More than 94% of cells are in the range $[0,31]$. However, there is more delay variation in the negative domain with RR algorithm than with ERR and FRRM algorithms, which have almost the same distribution.

LQF algorithm generally has the worst performance among observed algorithms, since its distribution is widest and it has more cells with large delay variations than other three algorithms.

VI. CONCLUSION

RR algorithm has by far the least amount of cells that have a zero delay variance. However, it shows the best performance, since the majority of cells are concentrated in a narrow range of the delay variation values, in comparison to other algorithms. It is followed by ERR and FRRM algorithms that have very similar characteristics.

FRRM algorithm has a bit better (with long buffers) or equal (with short buffers) performance compared to ERR algorithm in the case of IBP traffic. Under the uniform traffic these two algorithms have very close performances.

LQF algorithm shows the worst performance in all observed traffic conditions.

A general conclusion is that long buffers make the choice of algorithm less important regarding the maximal delay variance. RR and FRRM algorithms show the best performance if buffers are short.

However, *IPDV* is just one of the parameters that describe a switch performance. In order to have the whole picture of the switch performance, the other parameters for performance evaluation have to be observed.

Further research will be focused on the evaluation of switches with various sizes, as well as different traffic models.

REFERENCES

- [1] R. Rojas-Cessa, E. Oki, Z. Jing, and H. J. Chao, "CIXB-1: Combined input-one-cell-crosspoint buffered switch," in *Proc. IEEE HPSR '01*, Dallas, TX, USA, pp. 324–329, 2001.
- [2] Y. Kanizo, D. Hay, and I. Keslassy, "The crosspoint-queued switch," in *Proc. INFOCOM '09*, Rio de Janeiro, Brasil, pp. 729–737, 2009.
- [3] F. Abel, C. Minkenberg, I. Iliadis, T. Engbersen, M. Gusat, F. Gramsamer, and R. P. Luijten, "Design Issues in Next-Generation Merchant Switch Fabrics," *IEEE/ACM Transaction on Networking*, vol. 15, no. 6, December 2007, pp. 1603–1615.
- [4] M. Radonjić, and I. Radusinović, "Average Latency and Loss Probability Analysis of Crosspoint Queued Crossbar Switches," in *Proc. of 52nd International Symposium ELMAR-2010*, Zadar, Croatia, pp. 203–206, 2010.
- [5] M. Radonjic, and I. Radusinovic, "Impact of scheduling algorithms on performance of crosspoint-queued switch," *Annals of Telecommunications*, vol. 66, no. 5-6, May/June 2011, pp. 363–376.
- [6] M. Radonjić, I. Radusinović, I. Maljević, and D. Banović, "CQ switch analysis under the traffic overload," *Telfor Journal*, vol. 3, no.1, pp. 19–22, 2011.
- [7] M. Radonjic, and I. Radusinovic, "Buffer Length Impact to 32x32 Crosspoint Queued Crossbar Switch Performance," in *Proc. of 15th IEEE symposium on Computers and Communications (ISCC 2010)*, Riccione, Italy, pp. 954–959, 2010.
- [8] IP Packet Delay Variation Metric for IP Performance Metrics (IETF - RFC 3393). Available: <http://www.ietf.org/rfc/rfc3393.txt>
- [9] A. Mekittikul, "Scheduling Non-uniform Traffic in High Speed Packet Switches and Routers," Ph.D. thesis, Stanford University, November 1998.
- [10] E. Shin, V. Mooney, and G. Riley, "Round-robin arbiter design and generation," in *Proc. 15th International Symposium on System Synthesis*, Kyoto, Japan, pp. 243–248, 2002.
- [11] Y. Li, S. Panwar, and H. J. Chao, "Exhaustive Service Matching Algorithms for Input Queued Switches," in *Proc IEEE HPSR '04*, Phoenix, Arizona, USA, pp. 253–258, 2004.
- [12] D. Banovic, and I. Radusinovic, "Scheduling algorithm for VOQ switches", *AEU - International Journal of Electronics and Communications*, vol. 62, no. 6, pp. 455–458, June 2008.