The Dual Round Robin Matching Switch with Exhaustive Service

Yihan Li, Shivendra Panwar, H. Jonathan Chao

Abstract—**Virtual Output Queuing is widely used by fixed-length high-speed switches to overcome head-of-line blocking. This is done by means of matching algorithms. Maximum matching algorithms have good performance, but their implementation complexity is quite high. Maximal matching algorithms need speedup to guarantee good performance. Iterative algorithms (such as PIM and iSLIP) use multiple iterations to converge on a maximal match. The Dual Round-Robin Matching (DRRM) scheme has performance similar to iSLIP and lower implementation complexity. The objective of matching algorithms is to reduce the matching overhead for each time slot. In this paper we present the Exhaustive Service Dual Round-Robin Matching (EDRRM) algorithm, which amortizes the cost of a match over multiple time slots. While EDRRM suffers from a throughput below 100% for small switch sizes, it is conjectured to achieve an asymptotic 100% throughput under uniform traffic. Simulations show that it achieves high throughput under nonuniform traffic. Its delay performance is not sensitive to traffic burstiness, switch size and packet length. In an EDRRM switch cells belonging to the same packet are transferred to the output continuously, which leads to good packet delay performance and simplifies the implementation of packet reassembly.**

Keywords—**switching, scheduling, Virtual Output Queueing, Dual Round Robin, polling.**

I. Introduction

FIXED-LENGTH switching technology is widely accepted as an approach to achieve high switching efficiency for high speed packet switches. Variable-length IP packets are segmented into fixed-length "cells" at inputs and are reassembled at the outputs.

Packet switches based on Input Queuing (IQ) are desirable for high speed switching, since the internal operation speed is only slightly higher than the input line. However, an Input Queuing switch has a critical drawback [1, 2]: the throughput is limited to 58.6% due to the head-of-line (HOL) blocking phenomena. Output Queuing (OQ) switches have the optimal delay-throughput performance for all traffic distributions, but the N-times speed-up in the fabric limits the scalability of this architecture.

Virtual Output Queuing (VOQ) is used to overcome the drawbacks and combine the advantages of an Input Queuing switch and an Output Queuing switch. In a VOQ switch, each input maintains N queues, one for each output. By using VOQ, no additional speedup is required and HOL blocking can be eliminated.

Considerable work has been done on scheduling algorithms for VOQ switches. It has been proved that by using a *maximum weight matching algorithm* 100% throughput can be reached for i.i.d. arrivals (uniform or nonuniform) [3, 4, 5]. But maximum weight matching is not practical to implement in hardware due to its complexity, and may not guarantee fairness and quality of service. A number of practical *maximal matching algorithms* have been proposed [6, 7, 8], but maximal matching algorithm cannot achieve as high a throughput as maximum matching algorithms. Iterative algorithms such as PIM [9] and iSLIP [10, 5], use multiple iterations to converge on a maximal matching.

The Dual Round-Robin Matching (DRRM) switch [11, 12] builds and improves on the ideas incorporated in iSLIP. It has been proven that DRRM can achieve 100% throughput under i.i.d. and uniform traffic [12]. Furthermore, the DRRM scheme provides fairness and prevents starvation. It has lower implementation complexity compared to algorithms with similar performance and is scalable. According to simulation results [12], under uniform bursty traffic, the average delay of a DRRM switch varies approximately linearly with burst length, but under nonuniform traffic the throughput drops below 100%.

In this paper a variation of DRRM, Exhaustive service DRRM (EDRRM) is presented to improve switching performance under bursty and nonuniform traffic. The implementation of EDRRM and DRRM are comparable with both having lower complexity than iSLIP. The only performance drawback of EDRRM is that it does not achieve 100% throughput under uniform traffic for a range of switch sizes. We conjecture that for an EDRRM switch of large size, throughputs approaching 100% are achievable under uniform traffic. Simulation results presented in this paper and analysis results in [13] support, though not rigorously prove, this conjecture. For smaller switch sizes, we demonstrate this property through simulations. Compared to DRRM and iSLIP, EDRRM has higher throughput under nonuniform traffic. The delay of EDRRM is less sensitive to traffic burstiness, and increases much slower with switch size. EDRRM is neither a maximum matching nor a maximal matching algorithm. Unlike any maximum or maximal matching algorithm, which try to find as many matches as possible in each time slot, EDRRM achieves efficiency by looking at the matching overhead over time. In EDRRM the cost in wasted slot times to get a match may be large, but the cost is amortized over a VOQ busy period. We believe that this is a new approach with both theoretical and practical implications.

Most of the previous work only considers the cell delay that a cell suffers from the time it enters a VOQ to the time it is transferred to the destination output port. Additional delay is incurred at the Output Reassembly Module (ORM) of each output to reassemble packets [14]. Multi-

Yihan Li is a Ph.D. candidate in the Electrical and Computer Engineering Department, Polytechnic University, Brooklyn, NY 11201, email: yli@photon.poly.edu.

Shivendra Panwar and H. Jonathan Chao are on the faculty of the Electrical and Computer Engineering Department, Polytechnic University, Brooklyn, NY 11201, email: panwar@catt.poly.edu, chao@antioch.poly.edu

This work is supported in part by the New York State Center for Advanced Technology in Telecommunications (CATT), and also in part by the National Science Foundation under grants ANI0081527 and ANI0081357 .

ple queues are needed at each ORM if cells belonging to different packets are interleaved at the same output. When a cell is transferred through the switch fabric to the output, it is delivered to one of the queues of the ORM. The cells belonging to the same packet will be delivered to the same queue and can only leave the queue until the whole packet is reassembled. The total delay a packet suffers includes the cell delay and the time needed for reassembly. Thus the cell delay is not enough to evaluate the variable component of the delay incurred in a packet switch. In this paper we measure the packet delay as well as the cell delay of EDRRM and compare them to those of DRRM and iSLIP. We show that under uniform i.i.d. traffic, even when the cell delay of EDRRM is higher, the packet delay is lower, and it is not sensitive to switch size and packet length. At the same time, since all the cells belonging to the same packet are transferred to the output continuously, only one queue is needed in each ORM, which further simplifies the switch implementation.

In section II we briefly review the *DRRM* algorithm and its performance. In section III the EDRRM algorithm is described in detail. In section IV the throughput, average cell delay and average packet delay performance of an EDRRM switch is discussed.

II. The DRRM algorithm and its performance

In the DRRM scheme each input port maintains N VOQs. The algorithm has two steps:

Step 1 : Request. Each input sends an output request corresponding to the first nonempty VOQ in a fixed round-robin order, starting from the current position of the pointer. The pointer remains at that nonempty VOQ if the selected output is not granted in step 2. The pointer of the input arbiter is incremented by one location beyond the selected output if, and only if, the request is granted in step 2.

Step 2 : Grant. If an output receives one or more requests, it chooses the one that appears next in a fixed round-robin schedule starting from the current position of the pointer. The output notifies each requesting input whether or not its request was granted. The pointer of the output arbiter is incremented to one location beyond the granted input. If there are no requests, the pointer remains where it is.

The performance of the DRRM scheme has been shown and compared with that of iSLIP in [12]. Under uniform and i.i.d. traffic the throughput of a DRRM switch is 100%, and the average cell delay increases with switch size for a given load. The performance under nonuniform traffic is also considered in [12]. Simulation results show that under the hot-spot traffic, throughput for the hot-spot output of a DRRM switch is 100%. However, for nonuniform traffic scenarios, simulations show that the throughput of both DRRM and iSLIP drops below 100%.

III. The Exhaustive service DRRM scheme

In the DRRM scheme, when an input and an output are matched, only one cell is transferred from the input to the matched output. After that both the input and the output will increment their pointers by one and in the next time slot this input-output pair will have the lowest matching priority. This behavior is similar to the *limited service policy* [15] in a polling system. In order to improve on DRRM's performance under non-uniform traffic, we modified the DRRM scheme so that whenever an input is matched to an output, all the cells in the corresponding VOQ will be transferred in the following time slots before any other VOQ of the same input can be served. This is called the *exhaustive service policy* [15] in polling systems. We therefore call this the Exhaustive service DRRM (EDRRM) scheme.

In EDRRM, the pointers of inputs and outputs are updated in a different way from DRRM. In a time slot if an input is matched to an output, one cell in the corresponding VOQ will be transferred. After that, if the VOQ becomes empty, the input will update its arbiter pointer to the next location in a fixed order; otherwise, the pointer will remain at the current VOQ so that a request will be sent to the same output in the next time slot. If an input sends a request to an output but gets no grant, the input will update its arbiter pointer to the next location in a fixed order, which is different from DRRM where the input pointer will remain where it is until it gets a grant. The reason for this modification is as follows. In EDRRM if an input cannot get a grant from an output, it means that the output is most likely in a "stable marriage" with another input for all the cells waiting in the VOQ, and the unsuccessful input is likely to wait for a long time to get a grant from this output. It is better for the input to search for another free output than to wait for this busy one. Since an output has no idea if the currently served VOQ will become empty after this service, outputs will not update their arbiter pointers after cell transfer.

A detailed description of the two step EDRRM algorithm follows:

Step 1 : Request. Each input moves its pointer to the first nonempty VOQ in a fixed round-robin order, starting from the current position of the pointer, and sends a request to the output corresponding to the VOQ. The pointer of the input arbiter is incremented by one location beyond the selected output if the request is not granted in Step 2, or if the request is granted and after one cell is served this VOQ becomes empty. Otherwise, the pointer remains at that (nonempty) VOQ.

Step 2 : Grant. If an output receives one or more requests, it chooses the one that appears next in a fixed round-robin schedule starting from the current position of the pointer. The pointer is moved to this position. The output notifies each requesting input whether or not its request was granted. The pointer of the output arbiter remains at the granted input. If there are no requests, the pointer remains where it is.

Figure 1 shows an example of the EDRRM arbitration algorithm. r_1, r_2, r_3 and r_4 are arbiter pointers for inputs 1, 2, 3 and 4, and g_1, g_2, g_3 and g_4 are arbiter pointers for outputs 1, 2, 3 and 4. At the beginning of the time slot r_1 points to output 1 while g_1 does not point to input 1, which means that in the last time slot input 1 was not matched to output 1, and now input 1 requests output 1 for a new service. Similarly, r_2 requests output 3 for a new service. Since r_3 points to output 3 and g_3 points to input 3, it is possible that in the last time slot input 3 was matched to output 3 and in this time slot output 3 will transfer the next

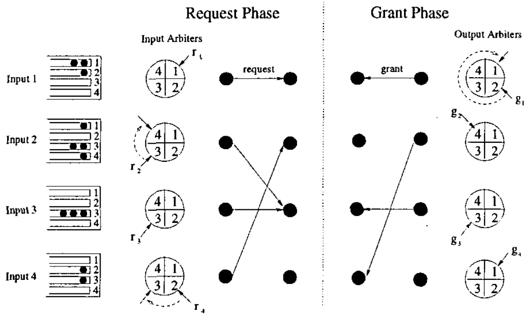

Fig. 1. An example of EDRRM

cell from input 3 because the VOQ is not empty. Input 4 and output 2 have a similar situation as input 3 and output 3. In the grant phase, output 1 grants the only request it receives from input 1 and updates g_1 to 1, output 2 grants the request from input 4 and output 3 grants the request from input 3. The request from input 2 to output 3 is not granted, so r_2 moves to 4. By the end of this time slot, the 1^{st} VOQ of input 1 and the 3^{rd} VOQ of input 3 are still nonempty so that r_1 and r_2 are not updated; r_4 is updated to 3 because the 2^{nd} VOQ of input 4 becomes empty.

The implementation complexity of EDRRM's switching fabric is identical to that of DRRM. Since the operational step and data exchange is limited, the DRRM arbitration mechanism can be implemented in a distributed manner to make the switch simpler and more scalable. The length of each control message in DRRM is only $\frac{1}{N}th$ of that in iSLIP. In [11] it is shown that by using a token-tunneling technique a switch capacity of more than one terabit/sec is achievable with existing electronic technology. The ORM of EDRRM is simpler than that of DRRM. Only one queue, with a buffer size equal to the maximum packet size, is maintained in the ORM of an EDRRM switch since cells belonging to the same packet are served sequentially from a VOQ. Usually, as in DRRM and iSLIP, since cells of different packets are interleaved, N queues are needed in each ORM, one for each input. In the next section, we will show that EDRRM has performance comparable with DRRM and iSLIP under uniform independent traffic, and has better performance under bursty traffic and nonuniform traffic.

IV. Simulated Performance of EDRRM

In this section the throughput and delay performance of EDRRM is shown by simulation results and compared with that of DRRM and iSLIP.

A. Throughput

DRRM is proved to have 100% throughput under uniform and i.i.d. traffic in [12]. According to simulation results the throughput of EDRRM under uniform and i.i.d. traffic is close to, but not quite 100%. Figure 2 shows the throughput of EDRRM with different switch size under uniform and i.i.d. traffic with 100% arrival rate. Note that the throughput first decreased and then increases with switch size. We conjecture that for larger N the throughput will approach 100% asymptotically. This conjecture is further supported by the analysis in [13]. Long simulation run lengths prevented us from considering switch sizes larger than 32. While this is certainly a weakness of EDRRM as compared to DRRM and iSLIP, we believe that this is an acceptable tradeoff given its performance advantages under more typical traffic loadings. We will consider these next.

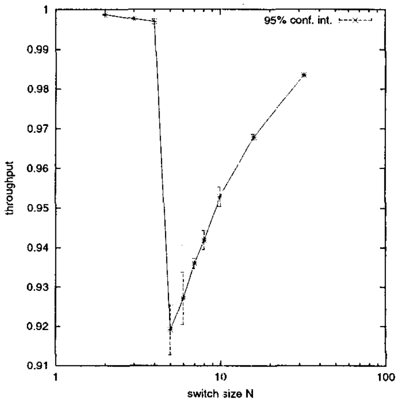

Fig. 2. The throughput of EDRRM switch under uniform traffic

In this paper we consider four nonuniform traffic patterns and compare the throughput of EDRRM to those of DRRM and iSLIP.

Pattern 1 : Hot-spot traffic, which refers to a traffic pattern where many inputs send traffic to one output line (the hot-spot) at the same time, thus overloading it. The hot-spot throughput of EDRRM is 100%, which is the same as that of DRRM and higher than that of iSLIP according to the results in [12].

Pattern 2 : In this pattern the arrival rates for all inputs are identical and equal to the loading of all outputs. For input i a fraction p, $\frac{1}{N} \leq p < 1$ of arrivals are destined to output i, and other arrivals are uniformly destined to other outputs. When $p = \frac{1}{N}$, this corresponds to the uniform case. When $p = 1$, all the arriving cells of input i are destined to output i.

Figure 3 compares the throughput of EDRRM, DRRM and iSLIP under this traffic pattern for different switch sizes. It shows that EDRRM has higher throughput than DRRM. The figure also shows that for the EDRRM switch, the throughput under uniform arrivals (when $p = \frac{1}{N}$) is comparable to the throughput under nonuniform arrivals (when $p > \frac{1}{N}$). In a DRRM switch each VOQ gets uniform service. On the other hand, in an EDRRM switch all the cells in a VOQ will be served when the VOQ obtains service no matter what the arrival rate is for this VOQ. When the system is stable, the service rate for a VOQ is close to the arrival rate, which leads to a high throughput. Note that another attractive feature of EDRRM is that it automatically

TABLE I

THE THROUGHPUT UNDER NONUNIFORM TRAFFIC PATTERN 3

f	0.1	0.2	0.3	0.4
EDRRM	0.969	0.949	0.931	0.872
DRRM	0.912	0.854	0.840	0.750
iSLIP	0.909	0.840	0.810	0.750

adapts to changes in traffic flow through the switch, i.e., no parameter has to be set to optimize performance for a given non-uniform traffic flow. This is important since it simplifies switch traffic management as traffic flows change with time. Also note that, as in Figure 3, the overall throughput first drops with switch size N and then increases with N. We see that with increasing switch size, the EDRRM switch can maintain close to 100% throughput for this nonuniform traffic pattern. Simulation results also show that the average cell delay of VOQs with heavy load is lower than that of VOQs with light load.

Pattern 3 : In this pattern the arrival rate for each input is the same. For input i a fraction f of arrivals are destined to output i, and other arrivals are destined to output $(i+1) mod N$. Table I compares the throughput of EDRRM, DRRM and iSLIP under this traffic pattern for different f with switch size 4. EDRRM has a higher throughput than DRRM or iSLIP. As f close to 0.5, the throughputs of DRRM and iSLIP are around 0.75. The reason is that both DRRM and iSLIP may be trapped in a bad state as shown in [16]. This did not happen to EDRRM in simulations.

Pattern 4 : In this pattern the arrival rate for each input is the same. From input i the traffic load to output $(i + j) mod N$ is two times the load to output $(i + j + 1) mod N$, $0 \leq j \leq N - 2$. Table II compares the throughput of EDRRM, DRRM and iSLIP under this traffic pattern for different switch sizes. The throughput of EDRRM is always above 90%, which is much higher than the throughputs of DRRM and iSLIP.

In traffic pattern 1, unfairness may occur for an EDRRM switch when one input occupies the hot-spot output for a long period and cells from other inputs destined to the hot-spot output cannot get through. To avoid unfairness, a limit on the maximum number of cells or packets that can be served continuously in a VOQ can be enforced by means of a counter. According to simulation results, with a VOQ cell service limit of 1000 cells, approximately $\frac{1}{N}$th of cells served by the hot-spot output come from each input. The performance of an EDRRM switch with a VOQ cell service limit does not differ much from the performance of EDRRM with no VOQ cell service limit for other patterns. We therefore do not include the simulation results of this variation in this paper. The variation can also help to balance the average delay between heavily loaded VOQs and lightly loaded VOQs in some level.

B. Average cell delay

Since the performance of iSLIP and DRRM are roughly comparable [12], we will compare the performance of EDRRM with DRRM from this point onward.

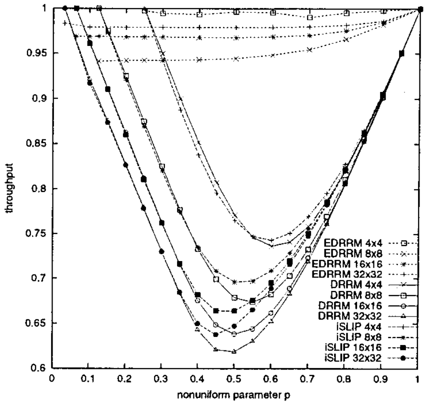

Fig. 3. The throughput for different switch sizes under nonuniform traffic pattern 2

TABLE II

THE THROUGHPUT UNDER NONUNIFORM TRAFFIC PATTERN 4

size	4	8	16	32
EDRRM	0.941	0.922	0.915	0.917
DRRM	0.756	0.722	0.720	0.720
iSLIP	0.745	0.720	0.719	0.719

Figure 4 shows the average cell delay of EDRRM and DRRM under uniform and i.i.d. traffic with different switch sizes. The average cell delay of an EDRRM switch under uniform traffic and a heavy load is larger than that of a DRRM switch. This is due to the more efficient DRRM scheduling mechanism under uniform, heavy traffic.

Figure 5 compares the average cell delay of an EDRRM switch and a DRRM switch with switch size of 16×16 under uniform and geometrically distributed bursty traffic with different average burst lengths. It shows that with the same average burst length, the average delay of DRRM is much larger than that of EDRRM under heavy load. The average delay of a DRRM switch increases approximately linearly with burst length, which is similar to the behavior of an EDRRM switch under light load. Significantly, under heavy load the average delay of an EDRRM switch does not change much with different average burst lengths and is much smaller than that of a DRRM switch for long burst lengths.

Figure 6 compares the average cell delays of EDRRM and DRRM switches with different switch sizes when the average burst length is 16 cells. We can see that as the switch size increases the average cell delay of a DRRM switch grows rapidly, while the average cell delay of an EDRRM switch grows more slowly. This indicates that cell delay for EDRRM is less sensitive to switch size than DRRM for bursty traffic.

The reason why EDRRM is not sensitive to bursty traffic is that all the cells arrive within the same burst will be served continuously so that little further delay will be suf-

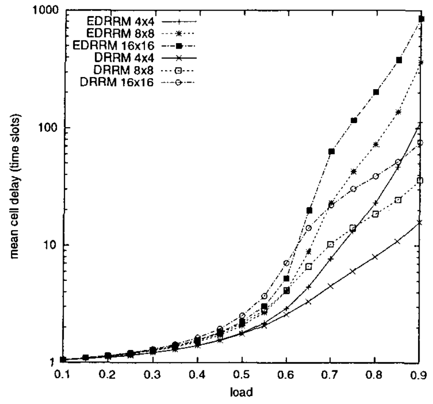

Fig. 4. The average cell delay under uniform traffic for different switch sizes

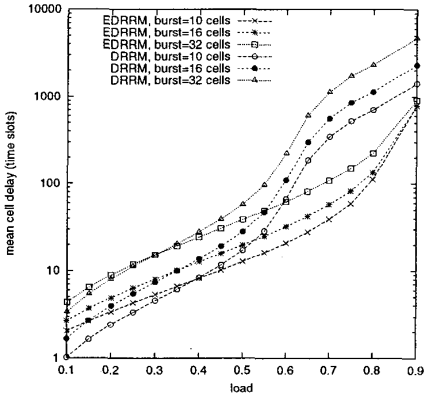

Fig. 5. The average cell delay under uniform and bursty traffic with different average burst lengths

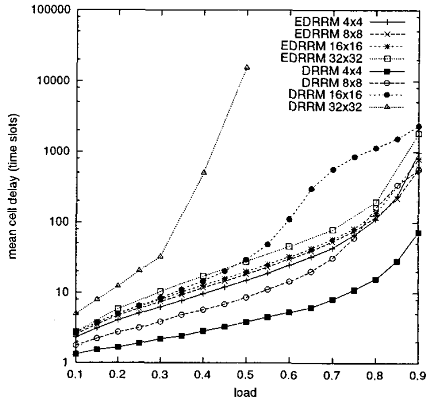

Fig. 6. The average cell delay under uniform and bursty traffic for different switch sizes

fered because of the burstiness. In DRRM or iSLIP, the service of cells in a burst is interleaved with service to other VOQ's, which may incur additional overhead due to unsuccessful requests. For cells belonging to a burst at a VOQ, when one cell is served, the next one cannot get service before all other nonempty VOQs at the same input have been served once.

C. Average packet delay

DRRM and EDRRM are fixed-length switching algorithms. Variable-length IP packets are segmented into fixed-length cells at the inputs, and the cells will be placed in the corresponding VOQ one by one. When a cell is transferred to its destination output, it will stay in a buffer and wait for the other cells in the same packet. After the complete reception of all the cells coming from the same packet, these cells will be reassembled into a packet. The delay a cell suffered before it is reassembled into a packet and delivered to its destination includes the cell delay discussed in the last subsection and the waiting time at the output reassembly buffer. So the cell delay performance is not sufficient to evaluate the packet delay performance of a fixed-length switch.

In this subsection, we consider average packet delay performance for the DRRM and EDRRM switches. After a packet is segmented into cells, one cell will be put into the VOQ in each time slot. As in [14], the packet delay of a packet is measured from the time when the last cell of the packet enters the VOQ until the time when the same last cell is transferred to its destined output. Simulation results show that the average packet delays of DRRM and iSLIP are similar to each other.

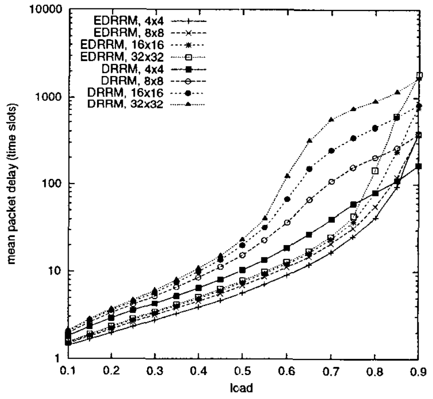

Fig. 7. The average packet delay of EDRRM and DRRM under uniform traffic with different switch sizes

Figure 7 compares the average packet delay of EDRRM and DRRM for different switch sizes under uniform i.i.d. Bernoulli traffic. The packet size is 10 cells. We can see that the average packet delay of EDRRM is always comparable with or smaller than that of DRRM when the switch size is larger than 4, even when the average cell delay of DRRM is smaller than that of EDRRM. The reason is that in DRRM, when one cell in a VOQ is served, the next cell in the same VOQ needs to wait for its next turn. For example, under heavy load when all the VOQs are nonempty, after a cell is transferred the next cell in the same VOQ needs to wait at least N time slots to get a service. For the cells in one packet, each cell needs to wait some time in the output reassembly buffer for the next cell except the last one. On the other hand, in EDRRM all the cells in the same packet will be served continuously, and the number of time slots that the packet needs for reassemble in an output buffer is the packet size in cells. Therefore the packet delay of an EDRRM switch is the cell delay for the first cell of the packet plus the packet size in cell slot times. Indeed, an EDRRM switch is really a packet switch and not a cell switch emulating a packet switch since packets are sent contiguously through the switch fabric. As a result, EDRRM does not need a complex reassembly buffer at each output port.

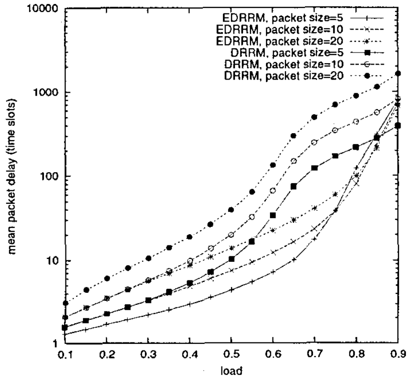

Fig. 8. The average packet delay with different packet sizes

Figure 8 shows the average packet delays of a 16x16 EDRRM switch and a DRRM switch with different packet sizes. We can see that for an EDRRM switch, under light load, longer packets suffer longer packet delay, while under heavy load the average packet delays for packets with different size are similar. On the other hand, in a DRRM switch the average packet delay increases linearly with the packet size. Though we do not present it here, delay performance for priority traffic (e.g. real-time traffic) can be enhanced by introducing a high and low priority queue in each VOQ.

V. Conclusions

The EDRRM algorithm is a variation of the DRRM scheduling algorithm. The implementation complexity of EDRRM's switching fabric is the same as that of DRRM, while packet reassembly is simpler than most other popular matching schemes. In an EDRRM switch when an input is matched with an output all the cells in the corresponding VOQ are served continuously before any other VOQ of the same input can be served. The average cell delay of an EDRRM switch can be analyzed by using an exhaustive random polling system model [13]. The performance of an EDRRM switch is comparable to or better than a DRRM switch or an iSLIP switch for most traffic scenarios. The only exception is the maximum throughput under uniform traffic. Under uniform i.i.d. traffic, an EDRRM switch has a larger average cell delay than a DRRM switch, but its average packet delay is lower and not sensitive to either switch size or packet size. Furthermore, EDRRM is not sensitive to traffic burstiness. Under nonuniform traffic the throughputs of a DRRM switch and an iSLIP switch drop well below 100%, while the throughput of an EDRRM switch is closer to 100%. To avoid unfairness under extreme traffic patterns such as "hot-spot", the maximum number of packets that can be served continuously in a VOQ can be bounded without changing the performance of EDRRM under other typical traffic scenarios.

References

[1] M. J. Karol, M. Hluchyj, and S. Morgan, "Input vs. output queuing on a space-division packet switch", *Proc. GLOBECOM* 1986, pp. 659-665.

[2] M. J. Karol, M. Hluchyj, and S. Morgan, "Input versus output queuing on a space-division packet switch," *IEEE Trans. on Communications*, vol.35, pp. 1347-1356, 1987.

[3] N. McKeown, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE INFOCOM'96*, pp. 296-302.

[4] N. McKeown, A. Mekkittikul, V. Anantharam and J. Walrand, "Achieving 100% throughput in an Input-Queued switch", *IEEE Trans. Communications*, vol. 47, No. 8, pp. 1260-1267, Aug. 1999.

[5] N. McKeown, "Scheduling algorithms for input-queued cell switches", *Ph.D. Thesis*, UC Berkeley, May 1995.

[6] A. Charny, P. Krishna, N. Patel and R. Simcoe, "Algorithms for providing bandwidth and delay guarantees in Input-Buffered crossbars with speedup", *IWQOS'98*, May 1998.

[7] P. Krishna, N. S. Patel, A.Charny and R. Simcoe, "On the speedup required for work-conserving crossbar switches", *IWQOS'98*, May 1998.

[8] A. Mekkittikul and N. McKeown, "A practical scheduling algorithm to achieve 100% throughput in input-queued switches", *IEEE INFOCOM 98*, Vol 2, pp. 792-799, April 1998.

[9] T. E. Anderson, S. S. Owicki, J. B. Saxe and C. P. Thacker, "High speed switch scheduling for local area networks," *ACM Trans. on Computer Systems*, vol. 11, No. 4, pp. 319-352, Nov. 1993.

[10] N. McKeown, "The iSLIP scheduling algorithm for Input-Queued switches", *IEEE/ACM Trans. Networking*, vol. 7, pp. 188-201, April 1999.

[11] H. J. Chao, "Saturn: a terabit packet switch using Dual Round-Robin", *IEEE Communication Magazine*, vol. 38 12, pp. 78-84, Dec. 2000.

[12] Y. Li, S. Panwar, H. J. Chao, "On the performance of a Dual Round-Robin switch," *IEEE INFOCOM 2001*, vol. 3, pp. 1688-1697, April 2001.

[13] Y. Li, S. Panwar, H. J. Chao, " Performance analysis of an Exhaustive Service Dual Round-Robin scheduling algorithm," *CATT Technical Report*, Nov. 2001.

[14] M. A. Marsan, A. Bianco, P. Giaccone, E. Leonardi, F. Neri, "Packet Scheduling in Input-Queued Cell-Based Switches," *IEEE INFOCOM 2001*, vol. 2, pp. 1085-1094, April 2001.

[15] H. Takagi, "Queueing anallysis of polling models: an update," *Stochastic Analysis of Computer and Communication Systems*, New York: Elsevier Science and B. V. North Holland, pp. 267-318 1990.

[16] C-S. Chang, D. Lee and Y. Jou, "Load balanced Birkhoff-von Neumann switches, part I: one-stage buffering ," *special issue of Computer Communications on "Current Issues in Terabit Switching,"* 2001.