Buffer Dumping Management for High Speed Routers

Caroline Fayet[1], André-Luc Beylot[2]

[1]INT, 9 rue Charles Fourier, 91011 Evry Cedex, France
caroline.fayet@int-evry.fr, Groupe des Ecoles des Télécommunications
[2]ENSEEIHT, 2, rue C. Camichel BP7122, 31071, Toulouse Cedex 7, France
andre-luc.beylot@enseeiht.fr, IRIT/TeSA Lab.

Abstract
To transport information with performance guarantees expressed in terms of delay, bounded delay variation, minimum packet loss, the architecture of multigigabit and terabit networks today designed in optical technology requires the use of high speed electrical/optical routers and switches. These interfaces need to manage efficiently the QoS, in particular they have to guaranty a short delay in the queues. This paper presents an efficient solution to optimize buffer dumping allowing to reduce significantly the delay or losses caused by buffer overflow. To highlight the proposed architecture we have analyzed its performance compared with different configurations of polling scheme.

Keywords: *buffer management, high-speed routers, QoS, performance analysis*.

1. Introduction

Routers implementing QoS functionalities requires some specificities. They need logical processes for the different traffics received, requiring many algorithms and also the availability of several queues for each output interface. A router making use of QoS functions needs to propose the following processes: classification, policing and marking, queuing and scheduling. First the classification of packets, in order to determine its output interface and which queue knowing its characteristics. We know that the classification may be done by informations contained inside the packet. Concerning the DiffServ model, the IETF has defined the DFCP field (Differentiated Service Code-Point) allowing to obtain 64 different values for a packet. Secondly policing and marking, to determine if the incoming traffic is conform to the foreseen service contract with a provider for example. In the negative, the packet may be discarded or tagged. Classification of incoming packets and control of traffic have been done to choice the appropriate queue, but in order to obtain an optimal process of traffics, it is mandatory to maintain a minimum occupancy of these queues. Loaded buffers presents two main drawbacks: they limit the availability of taking account of traffic burst which is a frequent phenomenon, moreover they increase the process delay inside the router causing an end to end delay often unacceptable. In order to reduce the queues length one way is to invite the transmitter to limit its throughput. In this kind of approach, it is necessary to remain that the congestion avoidance mechanism is just valuable for the mean time. An example can be done in frame relay using a bit BECN (Backward Explicit Congestion Notification), and FECN (Forward Explicit Congestion Notification). These bits invite the transmitter or the receiver to limit the flow. In the precise case of frame relay networks interconnecting routers the main problem consist to find routers able to understand these FECN and BECN bits in order to react consequently. This third step named “queuing” is composed, as explain previously, of congestion control and of an efficient dumping of the buffers of the routers. Concerning the short time only appropriated mechanisms related to the queues management of the used equipment will be efficient. Our work focus on this point. The last step is the scheduling, allowing to give the priority to the appropriate traffic.

High speed buffer dumping is the purpose of this paper. More precisely we propose a new design allowing to optimize the buffer dumping of the routers. A more detailed presentation of the proposed architecture is

described in [1]. Traditionally the buffers are emptied by means of a polling process, described in section 2, where all the queues are scanned and served sequentially even though for empty queues. The main drawback of processing empty queues result in a useless waste of time. The process polling used for many ATM switches correspond to this description [2] [3]. We find in the literature an improved solution for routers considering a reduced service time for empty queues. We propose a mechanism described in section 3 which offers two advantages, firstly a very efficient and simple solution by allowing the polling process to ignore empty buffers, secondly lowering the waiting time in non-empty queues. We present, in section 4, a performance analysis of this proposed solution, compared with the ATM-like polling and the classical polling scheme. We have focused on the polling process in order to overcome the weakness of this scheme. The following section presents two configurations of polling process most often used.

2. Polling processes

The polling process is the most common server model [4]. The system consists of sequentially scanning *N* queues and the server cyclically visiting all the queues as shown in figure 2. When visiting queue *i* then queue *i*+1, the "changeover time" is not equal to zero. Moreover, when visiting queue *i*, the server spends a significant amount of time even though it is empty [5]. As a consequence of this system the polling process is very efficient when every queue is non-empty and in case of uniform traffic. Unfortunately, this system which scans all the queues does not take account of empty queues which is a common case in the presence of traffic imbalances and of non-uniform traffic. Consequently we observe a waste of time due to the changeover time and the scanning of these empty queues, especially in case of unbalanced traffic. Two different cases have to be considered. Firstly the ATM-like polling where, as previously explained all the input queues are scanned and served with a constant cycle time even if there are empty. Secondly, the classical polling often studied in the literature in which all the input queues are scanned and served even if there are empty, but in this case the service time concerning empty queues corresponds to a reduced constant switch-over time. Unfortunately the time devoted to empty queues being not equal to zero, it increases the waiting procedure for priority information and obviously is a very bad solution for the probability of losing information.

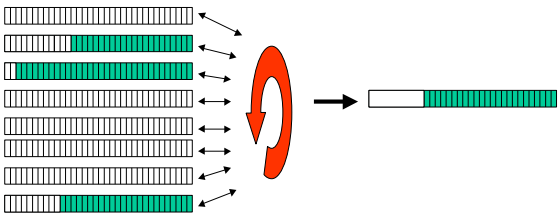

Figure 1: Polling Process System

Having described the drawbacks of these processes we propose in the following section a new design allowing to efficiently increase the performance of the buffer dumping.

3. Architecture of the buffer dumping management

The speed requirements put on broadband switches or routers impose a wired solution. A wired sequencer using digital circuits in a very high-speed technology available on the market today provides this hardware solution.
We consider the problem of multiplexing *N* input links to a single output. The situation occurs inside switching fabrics, high-speed routers or when matrices are interconnected in multistage switches. Most often (this is the case when multiplexing for concentration) the sum of the speeds on each input link exceeds the capacity of the output link, so that buffering is compulsory [7-8]. This amounts to consider *N* input links, which feed *N* input buffers. An overview of the global structure is shown in figure 2.

The proposed switching mechanism is composed of a *bus* and a register RI. This Data Bus will receive sequentially an

incoming information from the input registers. An incoming data located at the head of the elected queue is transferred to the bus, which carries it to the register RI where it is stored, waiting to be processed - i.e., redirected to the allocated output port.
The filtering of the incoming data is processed by means of very high-speed 3-state outputs.

Each input owns its proper wired state machine whose role is firstly to control the presence or not of the information in the associate register, then eventually its priority level and also the state of the other inputs. A decision is then taken enabling one input register among *N* to be loaded on the bus. This shows a constant interaction between the different computer blocks.

As opposed to classical polling schemes, here empty buffers are ignored. Classically, an empty buffer sends an empty information or cell which is overwritten later on. Therefore, non-empty buffers are visited more often, increasing significantly the performance of such a system.

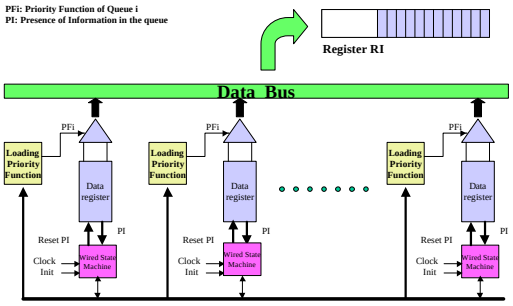

Figure 2: Block diagram of buffer dumping

3.1 Description of an Input Block

A selected input i has a two part structure:
- The data part, including the input data register and the 3-state outputs filtering the loading on the bus.
- The Local Control Unit, including the wired state machine whose role is to manage the different states and functions of the corresponding input, and the loading priority function allowing loading data on the bus.

3.2 Diversity of System functionalities

The main advantage of a such structure is to be adaptable to different kinds of processes.
Whatever the type of traffic, empty inputs are not taken into account. Therefore, they do not slacken the process. Consequently we have a gain in time, increasing significantly the performance of such a system.
The elementary choice is a sequential service scanning the inputs from the left to the right or the inverse. It will be possible to introduce a management of several level priorities per input if required.

The clock frequency is fixed by hardware and must be independent of the number of units. The clock frequency is 100 Mhz, in order to be in adequacy with the market.

4. Performance Analysis

Polling systems have been extensively studied during the last forty years. Many analysis were proposed [4-6] and more recently [12] for instance.

In the present paper, we will analyze and compare different configurations. A generic model with *N* queues with infinite waiting rooms served in cyclic order by a single server. Packets arrive to the queues according to a Poisson process with rates $l_1, l_2, \cdots, l_N$. The polling server has no initialization time. The service times of packets at queue *i* are constant with mean b_i and second moment $b_i^{(2)} = b_i^2$. All the considered configurations, presented in the previous parts corresponds to a *E-limited service* (limited from an exhaustive viewpoint) with a maximum of 1 packet served in a given cycle.

In order to consider unbalanced traffic patterns, we will consider that one input port has a higher input rate $l_1 = a\frac{l}{N}$ where *a* is a coefficient of asymmetry and $l = \sum_{i=1}^{N} l_i$ corresponds to the global input rate.

On the $N_2 = N-1$ other input ports, the input load is $l_2 = \frac{l}{N-1}\left(1-\frac{a}{N}\right)$

The following solutions are considered :

- ATM-like polling :

This first solution consists on considering that all the input queues are polled and "served" even if there are empty. This solution have been extensively studied in the ATM context for which periodic arrivals are considered and for which a constant cycle time is implemented with a duration equal to the minimal interarrival duration. The service time c_i will correspond to a constant switch-over time with mean s_i plus the service time b_i (a service time will be considered even if the input queue is empty), $c_i = s_i + b_i$. Its second moment is simply equal to $c_i^{(2)} = c_i^2$

- Classical polling :

In this solution, all the input queues are polled even if there are empty. In this case, the service time is equal to zero and a constant switch-over time with mean s_i and second moment $s_i^{(2)} = s_i^2$ is considered.

- Optimized Polling :

This solution corresponds to our optimized dumping management. Only non-empty queues are polled in a given cycle. For non-empty queues, the switch-over time is constant with the same parameters as those of the classical polling. As the initialization process duration is neglected, as long as the system remains empty no cycle are to be considered. In a given cycle, at least one packet is served, only non-empty queues are polled.

4.1 Analysis of the « ATM-like » polling

The analysis of the first case can be done through a $M/G/1$ queue analysis. Each input queue can be studied independently (a cycle will have a constant duration).

If a packet arrives in a non empty queue, its service time S_1 (long service time) will be equal to the duration of a cycle, $c = \sum_{i=1}^{N} c_i$

If the queue is empty, as Poisson arrivals are considered, its service time S_0 (short service time) will be uniformly distributed between c_i and $c + c_i$.

Let V (resp. B) denote the number of packets which arrives in a queue during a long service time (resp. short service time) (index i is omitted). Let $V(z)$ (resp. $B(z)$) be the z-transform associated to r.v. V (resp. B).

Let $\tilde{q}$ be the number of packets in the queue at steady state and $Q(z)$ its z-transform.

By writing Chapman-Kolmogorov equations and by summing over all the values of the number of packets in the queue, one can find

$$Q(z) = \frac{p_0(V(z) - zB(z))}{V(z) - z} \qquad (1)$$

where p_0 is the steady state probability for the queue to be empty.

Let $E[S]$ be the mean service time.

As,

$p_0 = 1 - l_i E[S]$ and

$$E[S] = p_0 E[S_0] + (1-p_0)E[S_1]$$

we obtain

$$p_0 = 1 - \frac{l_i E[S_1]}{1 - l_i E[S_0] + l_i E[S_1]}$$

In order to determine the mean number of packets in the queue $E[\tilde{q}]$, we can evaluate the derivative of those z-transforms.

Let $B^{(k)}(1) = \left.\frac{d^k B(z)}{dz^k}\right|_{z=1}$ and

$V^{(k)}(1) = \left.\frac{d^k V(z)}{dz^k}\right|_{z=1}$. By considering the derivative of (1) and using L'Hospital rule, one can find :

$$E[\tilde{q}] = p_0 \left\{ \frac{B^{(2)}(1) + 2B^{(1)}(1)}{2\left(1 - V^{(1)}(1)\right)} + \frac{B^{(1)}(1)V^{(2)}(1)}{2\left(1 - V^{(1)}(1)\right)^2} \right\}$$

The parameters $B^{(k)}(1)$ and $V^{(k)}(1)$ can easily be found :

$$\begin{cases} B^{(k)}(1) = \dfrac{l_i^k}{k!\,c}\left((c+c_i)^k - c_i^k\right) \\ V^{(k)}(1) = (l_i c)^k \end{cases}$$

The mean response time can be obtain by using Little's result :

$$E[R_i] = \frac{E[\tilde{q}]}{l_i}$$

4.2 Analysis of the « classical » polling solution

In the second case, the methods proposed in [9] and [10] can be applied as follows.

Let C_i denote the time between two consecutive visits to server *i*. $E[C_i]$ is independent of *i*, as long as the system is stable with a common value $E[C]$ with

$$E[C] = \frac{s}{1-r} \qquad (2)$$

where *s* is the mean switch-over time and r the total server utilization.

For the classical polling scheme, we obtain

$$s^C = \sum_{i=1}^{N} s_i \quad \text{and} \quad r^C = \sum_{i=1}^{N} r_i^C = \sum_{i=1}^{N} l_i b_i$$

Let $E[W_i]$ be the mean waiting time of queue *i*. It has been shown that in the 1-limited case (only a simplified version of these expressions is presented since the service times and the switch-over times are constant)

$$\sum_{i=1}^{N} r_i (1 - l_i E[C]) E[W_i] = A + E[C] \sum_{i=1}^{N} r_i^2 \quad (3)$$

where

$$A = \frac{r b^2}{2(1-r)} \sum_{i=1}^{N} l_i + \frac{rs}{2} + \frac{E[C]}{2}\left(r^2 - \sum_{i=1}^{N} r_i^2\right)$$

with $b = b_i$

Approximated methods were proposed to derive parameters $E[W_i]$. In the present paper, we implemented the methods initiated by [9] and [10]. It consists on considering a tagged customer. Its mean waiting time is equal to the mean residual time until the server next visits queue *i*, $E[q_i]$, plus the mean time necessary to serve the customers which are present in queue *i* when the customer arrives.

An approximate formula can be applied :

$$E[W_i] = \frac{\left(1 + \dfrac{r_i}{1-r}\right) E[q_i]}{1 - l_i E[C]}$$

One method consists on approximating $E[q_i]$ by a common value $E[q]$ which leads to a solution of (1) [10]. The second one [9] consists on considering that

$$E(C_{b,i}) E(q_i) = E(C_{b,j}) E(q_j)$$

where $E(C_{b,i}) = \dfrac{b+s}{1-r+r_i}$.

4.3 Analysis of the optimized polling

4.3.1 Introduction

For the optimized polling scheme, it is more convenient to include the switch-over time for non-empty queues in their service time which leads to a null switch-over time.

$$s^O = 0 \quad \text{and} \quad r^O = \sum_{i=1}^{N} r_i^O = \sum_{i=1}^{N} l_i (b_i + s_i)$$

The analysis of the Optimized Dumping solution is consequently more complicated because equation (2) can not be applied.

Nevertheless, a simple result can be applied to the whole system. The global mean response time and the global mean waiting are those of an $M/D/1$ queue with service time $d = b_i + s_i$ because no switch over time is lost. The difference between the actual system and the equivalent $M/D/1$ queue comes from the fact that in the proposed solution, the polling system may lead to a non-FIFO scheduling algorithm even if the scheduling is FIFO for each input queue.

The mean waiting time is consequently equal to

$$E[W] = \frac{l d^2}{2(1-r)} \qquad (4)$$

which is the mean waiting time when the traffic is balanced.

$$E[W_1] = E[W_2] = E[W] \quad (5)$$

where index 1 refers to the first queue and index 2 to the other ones.

The mean response time in all this analysis is simply equal to the mean waiting time plus the service time *d.*

It is a lower bound of the mean response time for the heavily loaded input queue and an upper bound of the mean response time for the other queues. This is due to the fact that the mean length of the first queue is larger than the other ones and consequently some packets of the other queues may overtake packets of the first queue.

4.3.2 Upper bound of the response time of the first queue

As the arrival processes are of Poisson type, the mean number of packets in the whole systems at arrival epochs are the same for all the input queues :

$$E[L] = \frac{r(2-r)}{2(1-r)}$$

Let us consider a packet *p* which arrives in the first queue. As this queue is heavily loaded, the number of packets in the other queues may be lower than the number of packets in the first queue. Consequently, most of the packets that are in the system when packet *p* arrives will leave the system before *p*. In the worst case, all the packets that enter all the other queues when *p* is waiting will also be served before *p*.

It leads to :

$$E[W_1^+] = E[W] + E[W_1^+] \sum_{i=2}^{N} l_i d$$

Finally,

$$E[W_1^+] = \frac{E[W]}{1-(N-1)l_2 d}$$

Using (4), it leads to a lower bound of the mean waiting time of the other queues :

$$l_1 E[W_1^+] + (N-1) l_2 E[W_2^-] = l E[W] \quad (6)$$

4.3.3 Lower bound of the response time of the first queue

The bound provided by (5) may be quite low especially when the input load is high. Another bound can be obtained by considering that the first queue is never empty and by analyzing the other queues.

The system can be analyzed like a symmetric 1-limited polling systems with a constant switch-over time $s = d$.

Equation (2) and (3) leads to

$$E[C] = \frac{d}{1-r'} \text{ with } r' = r - r_1$$

and

$$E[W_2^+] r' (1 - l_2 E[C]) = A + E[C] \frac{r'^2}{(N-1)}$$

with

$$A = \frac{r' d^2}{2(1-r')} (N-1) l_2 + \frac{r' d}{2} + \frac{E[C]}{2} r'^2 \frac{N-2}{N-1}$$

which can be simplified as follows :

$$E[W_2^+] = \frac{d}{2} \left(\frac{1 + (N-2) l_2 d}{1 - N l_2 d} \right)$$

Using (4), we finally derive :

$$l_1 E[W_1^-] + (N-1) l_2 E[W_2^+] = l E[W] \quad (7)$$

5 Results

In this part, we will present the validation of the models and the interest of the proposed scheduling method.

The following numerical values are considered :

- The number of input ports *N* is set to 8.
- $s_i = 1, b_i = 2, \quad 1 \le i \le N$

In the following figures, the mean response times will be expressed in number of clock time periods and the input rate in number of packets per clock time period.

Figures 3 and 4 correspond to the validation of the model for the Dumping Method mechanism. In figure 3 (resp. figure 4), we represented the mean response time of the first input port (resp. the other input ports) as a function the input load with a coefficient of asymmetry *a*=5.

R curves correspond to the mean response time in the symmetric traffic case, R- (resp. R+) to the lower bound (resp. upper bound). SIMUL results were obtained using discrete event simulations.

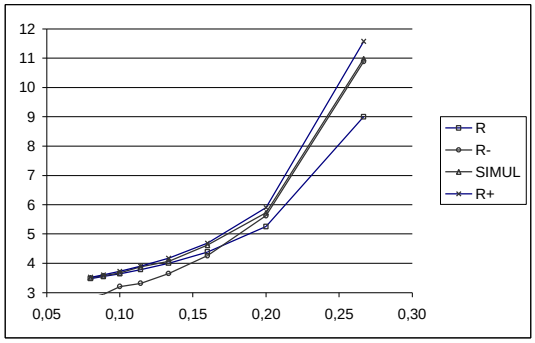

Figure 3. Mean Response Time as a function of the input load, first input port – Dumping Mechanism, *a*=5

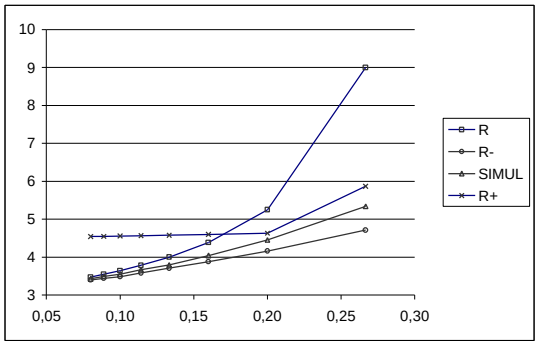

Figure 4. Mean Response Time as a function of the input load, other input ports – Dumping Mechanism, *a*=5

It is shown that the model leads to a good estimation of the mean response time. The lower bound of the mean response time for the first input port is really accurate even with high input rate. The difference between simulation results and this bound is about 1% to 2%. The upper bound is also really good, the difference is about 5%. When the input load is low, the bound given by the mean response time of the symmetric system is quite good.

For the other input ports, the bounds are also accurate ; the difference between the simulation results and the bounds are about 10%.

In figures 5 and 6, we represented the same performance criteria in the classical polling configuration. SIMUL curves correspond to the discrete event simulation results, BOXMA and WANG curves to the application of the Boxma [10] and Wang [9] approximations. It is shown that the approximations lead to accurate estimations of the performance criteria when the input load is not too high. In this configuration, the mean response time quickly increases with the input load especially for the first input port.

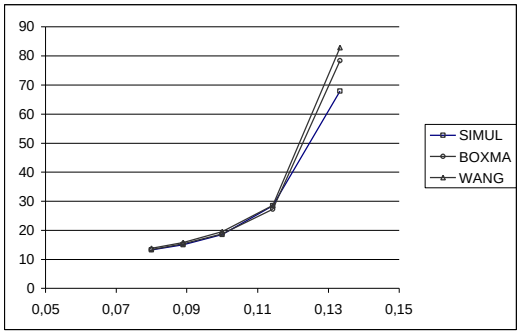

Figure 5. Mean Response Time as a function of the input load, first input port – Classical Polling, *a*=5

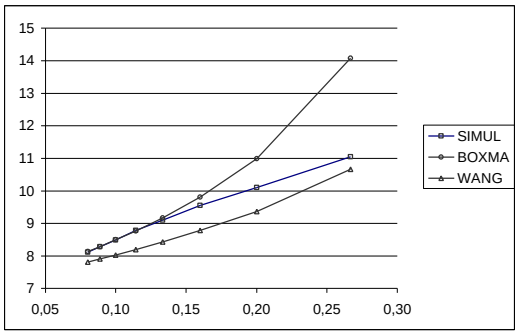

Figure 6. Mean Response Time as a function of the input load, other input ports – Classical Polling, *a*=5

In the following figures, we finally compare the results obtained with the ATM-like polling (exact results), the classical polling (Simulation results) and the optimized polling (Simulation results). When parameter *a* is equal to 5, the ATM-like solution is not stable for the first input queue. It is shown that the optimized dumping mechanism leads to really good performance results even when compared with the classical polling mechanism especially for the first input port.

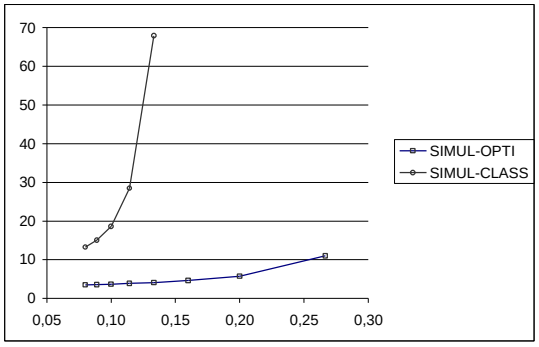

Figure 7. Comparison of the different mechanisms. Mean Response Time as a function of the input load, *a*=5, heavily loaded input port

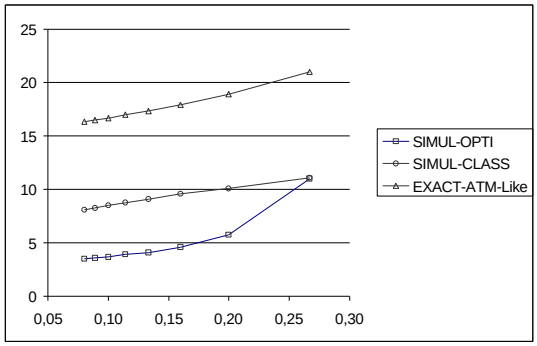

Figure 8. Comparison of the different mechanisms. Mean Response Time as a function of the input load, *a*=5, other input ports.

We finally compared the different mechanisms in the symmetric traffic case, for which we can derive exact analytical results for all the proposed configurations.

It is shown that the "optimized polling" mechanism leads to a great improvement of the system. The mean response time is lower for both heavy and low traffic. The improvement increases with the input load.

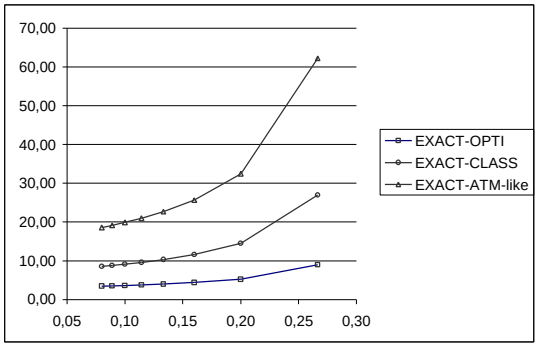

Figure 9. Mean Response Time as a function of the input load, symmetric traffic case.

5. Conclusion

In this paper, we proposed a new buffer dumping management mechanism for high speed routers. It consists on allowing the polling process to ignore empty buffers.

Analytical models were proposed to compare this solution to classical polling schemes. It has been shown, especially when considering unbalanced traffic patterns, that this mechanism improves the global performance of the system.

Prospective works deal with the comparison with other polling disciplines (gated or K-limited [12]) and the evaluation of the packet loss probability.

References

[1] C. Fayet, Buffer Management for Ring Node in the DAVID Metro Network, submitted to LCN'2003.

[2] S. Bush, R. Cruickshank, D. Bachar, F. Huang, Switching to ATM-The fore Runner ASX Network World- Feb. 1994.

[3] Catalyst 8500 Campus Switch Router Series, White paper, Cisco Systems, 1998.

[4] H. Takagi, Analysis of Polling Systems, Cambridge MIT Press, 1986.

[5] S. Borst, Polling Systems, Ph.D. Thesis, CWI, The Netherlands, 1994.

[6] H. Levy, M. Sidi, Polling Systems: applications, modeling and optimization, IEEE Transactions on Communications, Vol. 38, N° 10, October 1990, pp. 1750-1760.

[7] M. Karol, M. Hluchyj, S. Morgan, Input versus output queueing on a space division packet switch, IEEE Trans. Commun. 35 (1987) 1337-1356.

[8] M. Hluchyj, M. Karol, Queueing in high performance packet switching, IEEE J. Selected Areas Communications 6 (1988) 1587-1597.

[9] S.W. Furman, Y.T. Wang, Mean Waiting Time Approximations of Cyclic Service Systems with Limited Service, Performance'87, pp. 253-365, Elsevier Science.

[10] O.J. Boxma, B. Meister, Waiting-time approximations for cyclic-service systems with switch-over times, Performance Evaluation, Vol. 14, pp. 254-262, 1986.

[11] . T. Tedijanto, Non Exhaustive Policies in Polling Systems and Vacation Models: Qualitative and Approximate Approach, Ph.D Thesis, University of Maryland, 1990.

[12] T.L. Olsen, R.D. van der Mei, Periodic polling systems in heavy-traffic: distribution of the delay. Journal of Applied Probability 40, 1-22, 2003