AN EFFICIENT HIGH-SPEED PACKET SWITCHING WITH SHARED INPUT BUFFERS

E. Del Re, Senior Member IEEE
R. Fantacci, Senior Member IEEE

Dipartimento di Ingegneria Elettronica
Universita' di Firenze
Via S. Marta, 3 - 50139 Firenze (Italy)

ABSTRACT

This paper deals with an efficient high-speed packet switching in which each packet arrived at an input is stored in one of N possible queues, one for each possible output link. An implementation architecture which permits to share by the N separate queues the same input buffer is considered and studied. An important result shown in the paper is that the proposed multiple input queueing approach outperforms the output queueing approach without requiring a speed-up in the switching operations.

I. INTRODUCTION

The evolution in the field of communications which has been available advanced transmission systems using fiber optics, has led to advanced switching techniques able to handle multimedia traffic. The fast packet switching technique (FPS) seems to be a promising approach to be used in future high-speed networks. To highlight the main characteristics of the fast packet switching technique we note that every type of switch architecture must perform two basic function, i.e. routing and output contention resolution. The packet routing is usually based on hardware techniques by making use of the information contained in the header of each packet (usually called a cell). The solution of the output contention often represents the main source of complexity in the switch architecture [1]-[4]. It occurs whenever two or more packets arriving simultaneously at different switch inputs require to be routed to the same output. Only one of these contending packets achieves routing. In order to avoid loss, queueing is necessary for the others to wait for next routings. The two classic alternatives to queue the unrouted packets are the input queueing and the output queueing. Switching fabrics with input queueing are quite simple in architecture but unfortunately, the maximum possible throughput is bounded by 0.586 because of the head-of-line blocking problem [4]. Switches with output queueing avoid this drawback and achieve optimal delay throughput performance, in particular the maximum possible throughput approaches 1 as the mean arrival rate of packets per slot p approaches 1.

The main problem which arises with output queueing is the requirement of a faster switching fabric to route packets arriving at the switch inputs to the appropriate output buffers within a time slot therefore, by considering the worst case of N packets that simultaneously require to be routed to the same output the switch fabric has to operate N times faster than the input output links. It is evident that this requirement makes it difficult to use switching fabrics with output queueing in high speed networks. This paper deals with the input queueing approach, in particular to avoid the head of line blocking problem each input queue is splitted in N separate queues, one for each possible output link. Any arrival packet is stored in one of these queues according to its destination. Packets at the head of input queues for the same output link are contenders for routing. In this way it is possible to route to the outputs more than one packet for input queue, providing they have different output destinations. In this paper it is assumed that all the packets queued at each input share the same input buffer. In particular, it will be shown later that in this way it is possible to achieve the same throughput-delay performance as the output queueing approach without having to resort to a N time faster switch fabric and to reduce the buffer size requirements.

II THE MULTIPLE INPUT QUEUEING APPROACH

In the switch fabric under consideration whenever a new packet arrives at an input it is stored in the queue

The authors are with the Dipartimento di Ingegneria Elettronica, Università di Firenze, Via S. Marta,3 50139 Firenze, ITALY.

Work carried out under the financial support of the National Research Council (C.N.R.) in the frame of the Telecommunication Project.

associated to its output destination. In particular, we are focusing here on an implementation architecture (Fig. 1) in which all the queued packets at each input share the same buffer. Any new arrived packet is store in the shared input buffer (SB). The routing requests of such packets jointly with the memory locations are broadcasted over the input bus to all the output controllers (Arbiters). By means of routing request (address) filters (AF) a routing request may arrive only at the input of the arbiter corresponding to the desidered output as a routing request can only pass through the filter whose address matches the routing requests destination address. Collisions over the same bus are impossible because at almost one packet may arrived per slot at each input port. Each arbiter handles all the requests according to the First-In-First-Out (FIFO) selection policy. Therefore, a queue, named as destination queue, is formed by each arbiter and updated by placing at its end any new request.

From above, it follows that the fast packet switching is performed here in two stages. In stage one, the routing request, associated with each packet is analyzed while in stage two the packet itself is transmitted into the output link, whenever the associated routing request reaches the head of the appropriate destination queue. It is evident that in our model the routing requests may arrive in batches of random size. Looking to the worst case analysis, we may have to store N routing requests simultaneously within a time slot. However, the problem of a speed up typical of the output queueing implementation architecture, doesn't arise here because the routing requests are formed by few bits.

The performance analysis of the multiple input queueing approach (with shared input buffers) discussed above have been derived by making use of well-known results for discrete-time queueing system [11],[14]-[16]. Let us assume that the arrival processes on the N input links as N independent Bernoulli processes with the probability of an arrival per slot equal to p. Each packet has an equal probability to be addressed to any of the other N output links and successive packets require independent routing. Each destination queue, in its turn, may be modeled as a discrete $G^{(x)}/D/1/N/N$ queueing system. The goodness of this assumption will be verified in that follows by comparing theoretical and simulation results.

By means of the previous considerations it follows immediately that each input queue may be modeled as a discrete GEOM/G/1 queueing system with the service time per packet equal to the total delay spent by the corresponding routing request in the destination queue.

Fixing our attention on a particular input queue, the imbedded Markov chains approach developed for the continuous queueing system is applicable here also to derive the mean total delay per packet. The probability generating function of the number of packets in the input queue, assuming equilibrium, is:

$$Q(z)=\frac{Q_0A(z)(z-1)}{z-A(z)} \qquad (1)$$

where Q_o is the probability of having an idle queue and A(z) is the probability generating function of the number of arrivals during the service period of a customer. We have also [14],[17]:

$$A(z)=(1-p+pz)G(1-p+pz) \qquad (2)$$

where G(z) is the probability generating function of the waiting time (normalized to the packet duration time τ) or equivalently in our case the probability generating function of the total time spent by packets waiting for reaching the head of the destination queue.

In deriving an expression for G(z) it must be taken into account that in the considered case, the routing requests may arrive to the appropriate destination queue in batches of random size [14], [16],[17]. We assume that routing requests which arrive at the destination queue at a same instant are served in a random order. However, the routing requests arriving in earlier instants, are served first on the basis of the FIFO discipline. Therefore, the total time spent in the destination queue waiting for service by any routing request is due to the sum of two contributions, e.g. w_1 and w_2. The term w_1 takes into account the time necessary to serve all the routing requests which are waiting in the queue at the arrival instant. The second term w_2, is an additional delay due to the service of the routing requests which arrived at the same instant and were randomly selected to be served first.

Through an anlytical approach the mean delay per packet (normalized to the packet duration time τ) for the FIFO selection policy is derived in [17] as:

$$T=1+\sum_{k=0}^{N-1}\frac{(N+k-1)\alpha}{2}P_R(K)+p \left\{\frac{\sum_{k=0}^{N-1}k[k-1+\alpha(N-k-1)]P_R(k)}{2N-p[2+\sum_{k=0}^{N-1}(N-k-1)\alpha P_R(k)]}+\frac{\sum_{k=0}^{N-1}(N-k-)(N-k-2)\alpha^2P_R(k)}{3\{2NN-p[2+\sum_{k=0}^{N-1}(N+k-1)\alpha P_R(k)]\}}\right\} \qquad (3)$$

where $P_R(k)$ denotes the probability of having k routing request ($0 \le k \le N-1$) in a destination queue defined as:

$$P_R(1)=P_R(0)\frac{(1-a_{0,0}-a_{0,1})}{a_{1,0}} \quad (4)$$

$$P_R(k)=\frac{1-a_{k-1,1}}{a_{k,o}}P_R(k-1)-\sum_{i=2}^{k}\frac{a_{k-i,i}}{a_{k,0}}P_R(k-i) \quad (5)$$

$$2 \leq k \leq N-1$$

with $P_R(0)$ determined in order to verify the following equation:

$$\sum_{k=0}^{N-1} P_R(k)=1 \quad (6)$$

and the terms $a_{i,j}$, given by:

$$a_{i,j}=\binom{N-i}{j}\alpha^j(1-\alpha)^{N-i-j} \quad (7)$$

The paramiter α in (3), (7) can be obtained by solving numerically the following equation:

$$\alpha=\frac{p}{N-\sum_{n=0}^{N-1} nP_R(n)} \quad (8)$$

Fig. 2 shows T as a function of p for different values of N. It is evident in this figure that the maximum possible throughput approaches 1 as p approaches 1.

Fig. 3 shows T as a function of p for the single queueing on inputs and for the proposed multiple queueing on inputs in comparison with that obtained by using the output queueing approach [4]-[13]. It is evident in this figure that the proposed multiple queueing on inputs achieves the same performance as the output queueing approach without resorting to a more complex switching fabric [4],[18].

III FINITE BUFFER ANALYSIS

We can't ignore at this point that in any practical implementation, as that sketched in Fig.1, the buffers size are finite. It is evident that in this case packets may be loss. Unfortunately, for the switching system under consideration the analytical evaluation of the packet loss probability leads to a too complex queueing problem to be solved in a closed form. However, an approximation of the packet loss probability can be derived. The tightness of this approximation will be verified later by comparing analytical and simulation results.

We start our analysis by considering an input shared buffer of infinite size. Unfortunately, in the case of a shared input buffer the number of packets stored in each input queue is not independent of the number of packets stored in the other N-1 queues [19]. However, with the aim to simplify our analysis, we consider the overall number of packets stored in the input shared buffer as sum of N independent identically distributed (i.i.d.) random variables n_i, n_i denoting the number of packets stored in the queue for output i ($1 \leq i \leq N$). Note that this simplified approach is consistent with that outlined in [13] in deriving the performance of a switch fabric with a completely shared (output) buffering. It follows that the probability generating function of such number results to be:

$$B_t(z)=\sum_{k=0}^{\infty} z^k q_t(k)=B^N(z)=[\sum_{k=0}^{\infty} z^k q(k)]^N \quad (9)$$

with B(z) the probability generation function of the number of packets in one of the N separate input queues, q(k) the probability of having k packet in an input queue and $q_t(k)$ the probability of having k packets stored in the shared input buffer. In deriving an expression for B(z) we resort again to the imbedded Markov chain approach. The terms q(k) in (9) can be derived as:

$$q(i)=q(0)\frac{(1-a_0-a_1)}{a_1} \quad (10)$$

$$q(k)=\frac{1-a_1}{a_0}q(k-1)-\sum_{i=2}^{k}\frac{a_i}{a_0}q(k-i) \quad 2 \leq k \leq N-1. \quad (11)$$

with the terms a_i in (10), (11) are:

$$a_i=\sum_{m=1}^{\infty}\binom{m}{i}(p/N)^i(1-p/N)^{m-i}p(m) \quad (12)$$

In (12) p(m) denotes the probability of having for the packet at the head of the queue a service time equal to m (slots). The unknown term q(0) in (10), (11) can be defined by the standard conservation relation [5].
Therefore, it is possible to derive numerically the probability of having n packet stored in the shared input buffer. In our approach we approximate the packet loss probability P_B for a shared buffer of finite size equal to L (cells) as the probability of having stored a number of packet greater than L in a shared buffer of infinite size. Fig. 4 shows the derived approximation in comparison with the packet loss probability achieved by using the output queueing approach as a function of the buffer size

(cells) for different values of p. These figures clearly point out that the proposed multiple input queueing approach with an input shared buffer outperforms the output queueing approach. In Fig. 5 the our approximation is compared with the simulation results in the case of N=16. It is evident in this figure that our approximation is tight for low values of the packet loss probability, (tail of the distribution) which, typically, are the values of interest.

IV. CONCLUDING REMARKS

In this paper a fast packet switching fabric has been described and analyzed. The presented results clearly show that the multiple input queueing approach with input shared buffers outperforms the output queueing approach in terms of buffer size requirements. An important result is that the same delay-throughput performance as the output queueing can be achieved through the use of the proposed technique, without using a switch fabric which runs N times faster as the input and output links.

REFERENCES

[1] J. S. Turner, "Design of an Integrated Services Packet Network", **IEEE J. Select. Areas Commun.**, vol. SAC - 4, pp 1373-1380, Nov. 1986.

[2] J. S. Turner, L. F. Wyatt, "A Packet Network Architecture for Integrated Services", in **Proc. IEEE Globecom '83**, pp. 45- 50, Dec. 1983.

[3] P. Newman, "A Fast Packet Switch for Integrated Service Backbone Network", **J. Select Areas Commun.**, Vol. Sac - 6, pp. 1468-1479, Dec. 1988.

[4] H. Ahmadi, W. E. Denzel, "A Survey of Nodern High - Performance Switching Techniques", **J. Select. Areas Commun.**, vol. 7, pp. 1091 - 1103, Sept. 1989.

[5] L. Kleinrock, "**Queueing System**", vol. 1, New York, Wiley, 1975.

[6] J. F. Hayes, "**Modeling and Analysis of Computer Communications Networks**", New York, Plenum Press, 1984.

[7] M. Schwartz, "**Telecomunication Network: Protocols, Modeling and Analysis**", Reading, Massachusetts, U.S.A., Addison - Wesley Publishing Company, 1987.

[8] D. Bertsekas, R. Gallager, "**Data Network**", Engleewood Cliffs, New Jersey, Prentice Hall, 1987.

[9] J. L. Hammond, P. J. P. O'Reilly, "**Performance Analysis of Local Computer Networks**", Reading, Massachusetts, U.S.A., Addison - Wesley Publishing Company, 1986.

[10] A. S. Tanenbaum, "**Computer Networks**", Englewood Cliffs, NJ. Prentice - Hall, 1989.

[11] M.J. Karol, M.G. Hluchyj, S.P. Morgan, " Input Versus Output Queueing on a Space-Division Packet Switch", **IEEE Trans. on Commun.**, Vol. COM-35, NO. 12, pp. 1347-1356, Dec. 1987.

[12] J.Y. Hui, E. Arthurs, " A Broadband Packet Switch for Integrated Transport", **IEEE J. Select. Areas Commun.**, Vol.SAC-5, NO. 8, pp.264-1273, Oct. 1987.

[13] M.G. Hluchyj, M.J. Karol, " Queueing in High-Performance Packet Switching", **IEEE J. Select. Areas Commun.**, Vol. SAC-6, NO. 9, pp. 1587-1597, Dec. 1988.

[14] T. Meisling, " Discrete-Time Queueing Theory", **Oper. Res.**, Vol. 6, pp.99-105, Jan-Feb. 1958.

[15] H. Kobayashi, A.G. Konheim, " Queueing Models for Computer Communications System Analysis", **IEEE Trans. on Commun.**,Vol. COM-25, NO. 1, pp. 2-28, Jan. 1977.

[16] P.J. Burke, "Delays in Single-Server Queues with Batch Input", **Oper. Res.**, Vol. 23, pp. 830-833, July-Aug. 1975.

[17] E. Del Re, R. Fantacci, "A Fast Packet Switching Satellite Communication Network", IEEE INFOCOM'91, Miami, Florida, U.S.A., April, 7-8 1991.

[18] J.Y. Hui, "**Switching and Traffic Theory for Integrated Broadband Networks**", Kluwer Accademic Publishers, Norwell, Massachutes, U.S.A., 1990.

[19] A.E. Eckberg, T-C. Hou, "Effect of Output Buffer Sharing Requirements in an ATDM Packet Switch", IEEE INFOCOM'88, pp. 459-466.

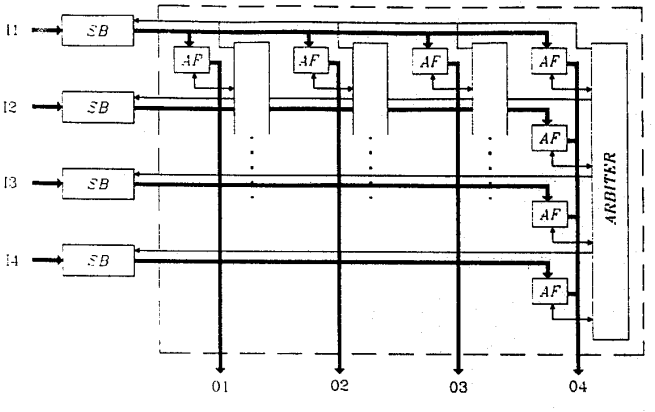

Fig. 1 - The proposed switch fabric architecture (N=4).

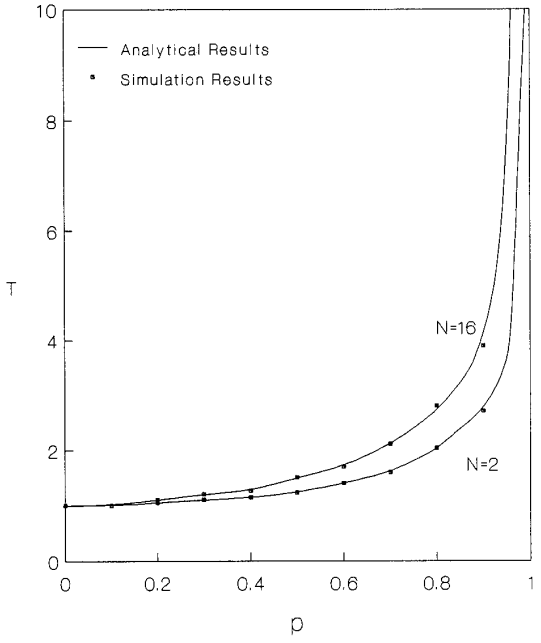

Fig. 2 - The mean normalized total delay

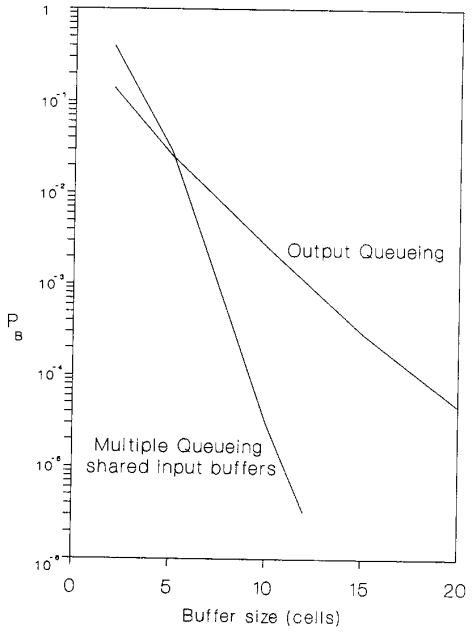

Fig. 4 - Packet loss probability (N=32;p=0.8)

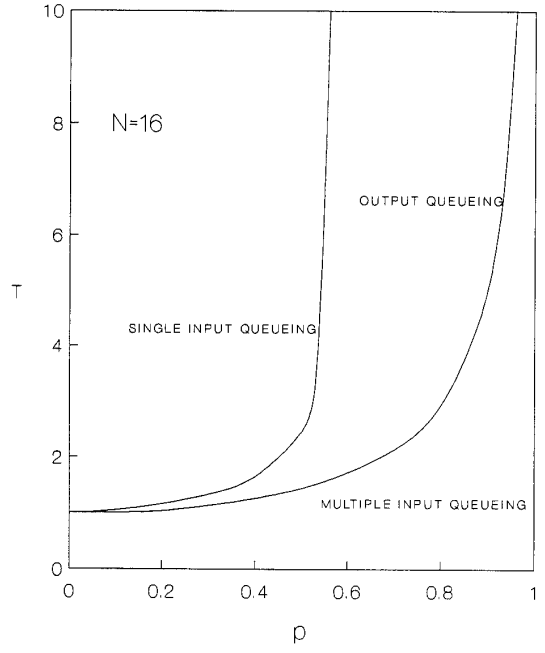

Fig. 3 - Mean normalized total delay comparison

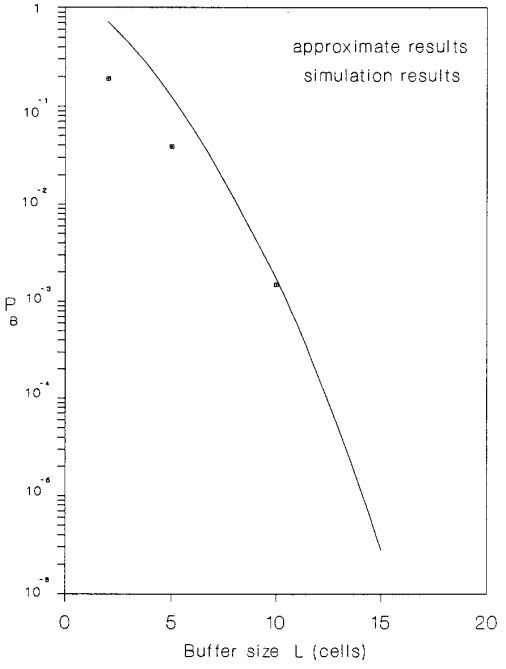

Fig. 5 - Packet loss probability comparison (N=16;p=0.9)