Quasi-Pushout Cell Discarding

Yu-Sheng Lin and C. Bernard Shung

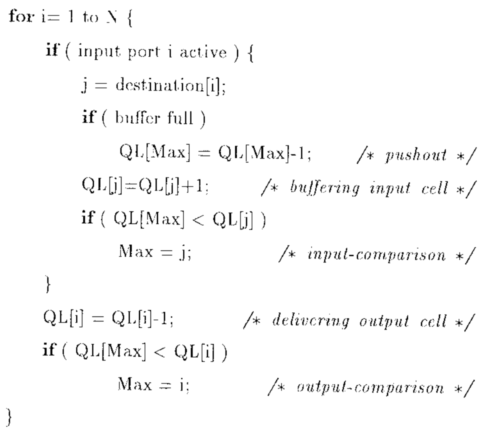
***Abstract*— Cell discarding takes place when the buffer space of a network node is used up. Though *Pushout* cell discarding was found to achieve fair buffer utilization and good cell loss performance, it is difficult to implement because of the large number of queue length comparisons. In this letter, we propose *Quasi-Pushout* cell discarding which reduces the number of queue length comparisons by employing the concept of quasi-longest queue. Simulation results under bursty and imbalanced traffic conditions show that Quasi-Pushout can achieve comparable cell loss performance as Pushout at a much lower complexity.**

***Index Terms*—Cell loss performance, pushout discarding,**

I. INTRODUCTION

IN ASYNCHRONOUS transfer mode (ATM) switching networks, buffers are required to accommodate traffic fluctuations due to statistical multiplexing. However, *cell discarding* may take place at a network node when the buffer space is used up during a traffic surge. The well-known *Pushout* (PO) cell discarding has been shown to offer optimum cell loss performance [1], [2]. When the buffer is full, the PO scheme discards one cell in the longest queue to make room for the incoming cell. Despite the optimum performance, PO is very difficult to implement because it requires $O(N)$ queue length comparisons to find out the longest queue, where N is the number of output queues. When N is large, these comparisons may become the speed bottleneck. Other *threshold-based* cell discarding schemes, which keep the input cells from entering the over-threshold queues, were easier to implement while providing a sub-optimum performance [3], [4]. The main drawback of these threshold-based schemes is *nonspace-conserving*, i.e., cell discarding occurs *before* the buffer is full.

In this letter, we propose the *Quasi-Pushout* (QPO) cell discarding, which features a much reduced hardware complexity than PO. An index Max for the quasi-longest queue is maintained and updated during cell arrival or departure events. At the time when buffer is full, one cell is discarded from the quasi-longest queue to make room for the incoming cell. A shared buffer ATM switch was simulated under bursty and imbalanced traffic conditions to compare cell loss performance of the proposed QPO and other cell discarding schemes.



```
for i= 1 to N {
    if ( input port i active ) {
        j = destination[i];
        if ( buffer full )
            QL[Max] = QL[Max]-1;        /* pushout */
        QL[j]=QL[j]+1;        /* buffering input cell */
        if ( QL[Max] < QL[j] )
            Max = j;        /* input-comparison */
    }
    QL[i] = QL[i]-1;        /* delivering output cell */
    if ( QL[Max] < QL[i] )
            Max = i;        /* output-comparison */
}
```

Fig. 1. The QPO scheme makes queue length comparisons on the arrivals and departures of cells to track the quasi-longest queue Max.

II. QUASI-PRODUCT CELL DISCARDING

Fig. 1 shows the algorithm of the proposed QPO cell discarding scheme for a shared buffer ATM switch. For each port i, if the incoming cell is active and the buffer is full, the quasi-longest queue Max will discard one cell to make space for the input cell. In contrast to PO which needs N comparisons to determine the real-longest queue for every discarded cell, QPO *tracks* the quasi-longest queue by using two comparisons only. One is on the arrival of an input cell: the queue length of the destination queue j is increased and compared with that of queue Max. The other is on the departure of an output cell: the queue length of the output queue i is decreased and compared with that of queue Max. If the new length of queue i or j is longer than that of queue Max, the index Max is redirected to the new quasi-longest queue.

Let us analyze the difference between the quasi-longest queue and the real-longest queue. Without loss of generality, we can assume that initially the quasi-longest queue Max, and another queue k, are both the longest queues. When the buffer is full or when $i = Max$ (i.e., queue Max is the output queue), the queue length of Max will decrease such that temporarily the quasi-longest queue is no longer the real-longest one. This situation will be corrected when queue k is served, either as an output queue or a destination queue. Therefore, the proposed algorithm may produce sub-optimum results due to mistracking the longest queue occasionally. But from the simulation results to be shown in the next section, the performance of the proposed algorithm is very close to that of the optimum PO algorithm.

Note that there are two queue length comparisons in QPO for each port processed, which is still the double of threshold-

---

Manuscript received April 11, 1997. This work was supported by National Science Council of the Republic of China under Grant NSC-84-2215-E-009-046. The associate editor coordinating the review of this letter was Prof. C. Douligeris.

The authors are with the Department of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C. (e-mail: yslin@ee.nctu.edu.tw).

Publisher Item Identifier S 1089-7798(97)07471-1.

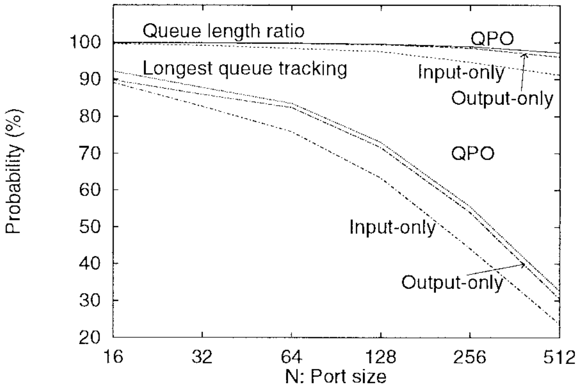


Fig. 2. The tracking probability and queue length ratio between the quasi-longest queue and the real-longest queue of QPO schemes with different switch dimensions.

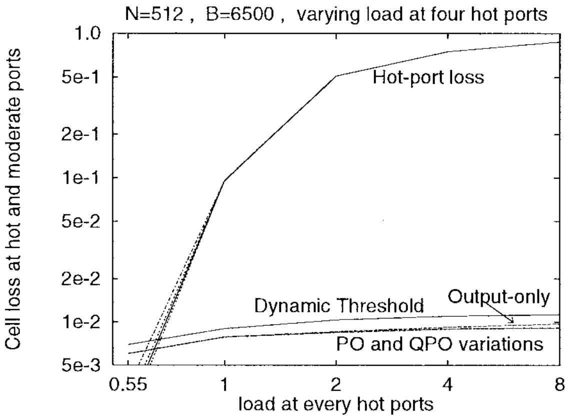


Fig. 3. Cell loss performance of QPO schemes with imbalanced traffic and varying load of four hot-spot output ports.

based schemes. Should it be desired to further reduce the computation, one of the two comparisons can be omitted. We will refer to these two variations as *output-only* QPO, which only compares the output queue with the quasi-longest queue, and *input-only* QPO, which only compares the destination queue with the quasi-longest queue. It is intuitively obvious that fewer comparisons result in longer period of suboptimal situation. For output-only QPO, each queue is served periodically so the sub-optimum duration is upper-bounded. For input-only QPO, hot-spot ports can be easily tracked by their frequent cell arrivals. The performance of all QPO schemes is simulated and shown in the following section.

III. SIMULATION RESULTS

An $N \times N$ shared buffer ATM switch with a B-cell buffer using different discarding schemes was simulated under bursty and imbalanced traffic. The bursty traffic was generated using the active/silent model with exponentially distributed burst length [5]. There were h hot-spot ports with L_h load, while other $(N - h)$ moderate ports were loaded at 0.55. If not explicitly stated, the simulation was performed for a period of 10^7 cell departures, with $N = 512, B = 6500, L_h = 0.95, h = 4$, and the active burst length being 20.

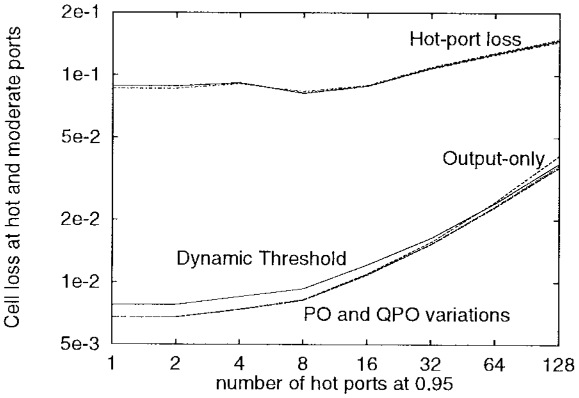


Fig. 4. Cell loss performance of QPO schemes with imbalanced traffic of varying numbers of hot-spot output ports at 0.95 load.

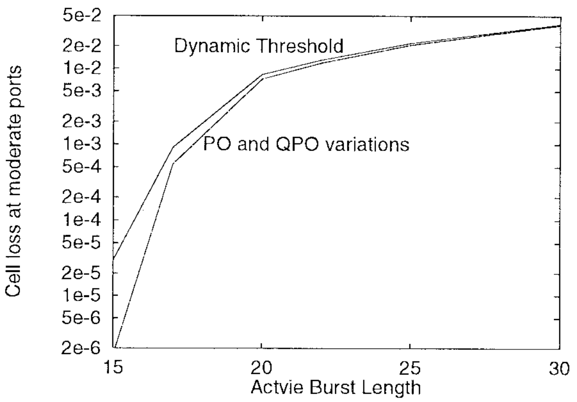


Fig. 5. Cell loss performance of QPO schemes with different burst length.

First we illustrate the capability of tracking the real-longest queue by QPO for different switch dimensions in Fig. 2. On every cell discarding event, all queues are sorted by length to see if the quasi-longest queue is tracking the real-longest one. The queue length ratio of the quasi-longest queue to the real-longest queue is calculated to show the severity of sub-optimal (mis-tracking) condition. For all QPO scheme, the capability of tracking the longest queue degrades as N grows, which reflects the tradeoff with the comparison cost (from N to 2 or 1). However, we found, for all QPO schemes, the average queue length ratio is above 0.9 even for large N. In other words, though QPO schemes cannot track the real-longest queue precisely for large N, the length of the quasi-longest queue is still very close to that of the real-longest queue.

As the two variations of QPO are considered, the input-only scheme has lower tracking capability because the randomly distributed cell destinations will not guarantee a complete check of all queues. On the other hand, the output-only scheme, which periodically checks all queues, has about the same tracking capability as QPO. But, if the load L_h or the number of hot-spot ports h is increased, cell arrivals on these hot-spot ports become more frequent and the input-only scheme may become more favorable than the output-only scheme. Such effects are demonstrated in the following cell loss performance comparisons among the proposed QPO schemes, PO, and the dynamic queue length threshold scheme [4].

Fig. 3 shows the cell loss performance of a switch with four hot-spot output ports and the hot-spot load varying from 0.55 to 8. All QPO schemes have about the same cell loss as the PO scheme, while the nonspace-conserving dynamic threshold scheme has higher cell loss. As the load to the hot-spot ports grows over unity, the output-only scheme cannot react to such overload, and may suffer a slightly higher cell loss than PO and QPO. On the contrary, the input-only scheme checks the hot-spot ports more frequently, thus achieving as good cell loss as PO and QPO. This difference can also be seen in Fig. 4, which shows the cell loss versus different numbers of hot-spot ports with 0.95 load.

Beside for various hot-spot conditions, the cell loss difference between all QPO schemes and PO is not significant for varying burst length and buffer size. The simulation result of varying burst length from 15 to 30 is shown in Fig. 5. All QPO schemes, despite their different tracking capabilities, provide performance comparable to PO, and the difference is indistinguishable.

IV. SUMMARY

In this paper, we proposed the QPO cell discarding which greatly reduce the number of queue length comparisons by tracking the quasi-longest queue instead of sorting out the real-longest queue. We verified through simulations that QPO offers comparable cell loss performance as the optimum Pushout scheme.

REFERENCES

[1] I. Cidon, L. Georgiadis, and R. Guérin, “Optimal buffer sharing,” in *Proc. INFOCOM*, 1995, pp. 24–31.

[2] L. Tassiulas, Y. C. Hung, and S. S. Panwar, “Optimal buffer control during congestion in an ATM network node,” *IEEE/ACM Trans. Networking*, vol. 2, pp. 374–386, Aug. 1994.

[3] J. F. Meyer, S. Montagna, and R. Paglino, “Dimensioning of an ATM switch with shared buffer and threshold priority,” *Computer Networks and ISDN Syst.*, no. 26, pp. 95–108, 1993.

[4] A. K. Choudhury and E. L. Hahne, “Dynamic queue length threshold in a shared memory ATM switch,” in *Proc. INFOCOM*, 1996, pp. 679–687.

[5] S. C. Liew, “Performance of various input-buffered and output-buffered ATM switch design principles under bursty traffic: Simulation study,” *IEEE Trans. Commun.*, vol. 42, pp. 1371–1379, Feb./Mar./Apr. 1994.