QLP: A Joint Buffer Management and Scheduling Scheme for Input Queued Switches

Dequan Liu, Nirwan Ansari and Edwin Hou
Advanced Networking Laboratory
Department of Electrical and Computer Engineering
New Jersey Institute of Technology, Newark, NJ
dxl0211@oak.njit.edu, {ansari,hou}@njit.edu

***Abstract*- In this paper, we propose the Queue Length Proportional (QLP) assignment algorithm for input queued switches that considers buffer management and scheduling mechanism inclusively to obtain an optimal assignment of both bandwidth and buffer space according to the real traffic load. The bandwidth assignment is implemented by considering both bandwidth and backlogged queue lengths, so that it is possible to obtain a high throughput as well as a low cell loss ratio at the same time. QLP is shown to be able to maximize overall throughputs and improve buffer utilization as compared to those which treat buffer management and scheduling as separate functions.**

Index Terms: Traffic scheduling, buffer management, input queued switches.

I. INTODUCTION

Many switching architectures have been considered for Asynchronous Transfer Mode (ATM) networks [1,2]. Depending on the position of the buffer, a switch can be classified as input, output, and input-output queued one.

How to efficiently allocate the buffer space has been extensively studied [3,4,5]. Complete Sharing (CS) [3] policy allows cells enter into the buffer until it is full. This policy can perform very well under light load, but it can cause severe unfairness under asymmetrical or heavy loading condition because heavy connections can occupy the whole buffer and starve other connections. Complete Partitioning (CP) [3] policy, on the other hand, divides the buffer into separate sections, and each of them can be accessed only by a particular connection. Fixed or dynamic threshold can be assigned to each connection. If one connection reaches its threshold, cells from this connection are not allowed to enter into the buffer. This policy guarantees fairness among all connections but may incur high cell loss ratio. Push-out is a technique to support multi-priorities and fairness by replacing existing cells with new ones when the buffer is full. It is shown in [4,5] that Push-out with Threshold (POT) is the optimal policy in terms of cell loss ratio.

Although output queued switches can provide QoS guarantees, they are limited by the speedup requirement---the processing speed of data line inside the fabric and the rate to access the buffer should be N times the outside line rate for an N by N switch. In high-speed networks, with high-speed optical fiber being the transmission media, this requirement is becoming much more difficult to be satisfied.

The fabric and the memory of input queued switches, on the other hand, can run at the same rate as the outside line. The well known Head-of-Line (HOL) blocking problem can be eliminated simply by using Virtual Output Queuing (VOQ), where each input maintains a separate virtual queue for each output. In this paper, each input port is assumed to have its own buffer.

A key issue related to input-queued switches is scheduling cells to obtain a high throughput as well as a low cell loss ratio. Mckeown *et al.* [6] presented a mechanism to achieve up to 100% throughput by finding a matching of a bipartite graph during every time slot. This algorithm performs very well when the traffic is admissible. However, the computational complexity is $O(N^{2.5})$ per time slot. Recently, a novel algorithm proposed by Chang *et al.* [7] can guarantee not only a high throughput but also a bounded delay. The matching of a bipartite graph is computed over many time slots (e.g., 1000 time slots) rather than one time slot. Thus the new algorithm is "good on average" with much lower computational complexity per time slot. The proposed algorithm is modified from and adopts many advantages of Chang *et al.*'s algorithm. Other scheduling algorithms for input-queued switches such as the ones in [8,9,10,11] can also achieve a high throughput. However, these scheduling algorithms implicitly assume that the buffer space inside a switch is large enough which may be a limiting factor in practice.

Lapiotis and Panwar [12] showed that joint buffer management and service scheduling for output-queued switches can improve the utilization of switch resources better and accommodate more traffic in the network. We propose to adopt this joint optimization concept for input queued switches, which are scalable, resulting in the Queue Length Proportional (QLP) assignment algorithm. Here, we focus on the condition when overloaded traffic lasts for a long enough time. It is shown that appropriate joint assignment of both buffer space and bandwidth according to the real traffic load will lead to not only a high throughput, but also a low cell loss ratio. An intuitive explanation for this provision is that there is no benefit to assign more bandwidth to a connection than its assigned buffer space can accommodate. On the other side, it is not necessary to assign more buffer space to a connection with low allocated bandwidth (rate), especially under the heavy traffic condition.

Guerin *et al.* in [13] presented a special scheme, on another extreme direction, to provide QoS features through buffer management only.

This paper is organized as follows: in section II, we review the Birkhoff-Von Neumann algorithm presented in [7]. The novel integrated method QLP along with analysis is introduced in section III. Simulation results are given in section IV.

II. BANDWITCH ASSIGNMENT

To show how QLP (which concentrates on how the leftover bandwidth can be efficiently allocated to the best effort traffic) can work together with the Birkhoff-Von Neumann algorithm (which provides enough bandwidth to the guaranteed traffic), we will first review the Birkhoff-Von Neumann algorithm in this section.

Let $\lambda = (\lambda_{i,j})_{N\times N}$ be the rate matrix of a switch with N input ports and N output ports. Here $\lambda_{i,j}$ denotes the rate demand from input i to output j. It is said to be *non-overbooking* if the following two inequalities are satisfied:

$$\sum_{i=1}^{N} \lambda_{i,j} \le 1 \quad j = 1,\dots,N \tag{1}$$

$$\sum_{j=1}^{N} \lambda_{i,j} \le 1 \quad i = 1,\dots,N \tag{2}$$

Then, there exists a set of positive coefficients C_K and associated permutation matrices M_k, $k=1,\dots,K$ (K is the decomposition number which is less than N^2-2N+2) that satisfy:

$$\lambda \le \sum_{k=1}^{K} C_k M_k, \quad and \quad \sum_{k=1}^{K} C_k = 1.$$

After obtaining the coefficients and the permutation matrices, we can set the connection of a switch according to the permutation matrices with the connection duration proportional to the relative coefficients.

A. The converting algorithm: algorithm 1 in [7]

The rate matrix λ is called doubly substochastic if it satisfies conditions (1) and (2). If both (1) and (2) are equalities, then the matrix is called doubly stochastic.

Algorithm 1 derives a doubly stochastic matrix R from the doubly substochastic matrix λ.

Step 1: Randomly find an element at (i,j) position in λ satisfying $\Sigma_j \lambda_{i,j}<1$ and $\Sigma_i \lambda_{i,j}<1$.

Step 2: Let $\varepsilon=1-max[\Sigma_j \lambda_{i,j}, \Sigma_i \lambda_{i,j}]$. Then add ε to the element at (i,j) in λ.

Step 3: Repeat step 1 and 2 until the sum of all elements in λ is equal to N.

B. The decomposition algorithm: algorithm 2 in [7]

Let R be the doubly stochastic matrix derived from the original doubly substochastic λ by algorithm 1 and $(i_1,\dots, i_N)$ be a permutation of $(1,\dots,N)$ satisfying:

$$\prod_{k=1}^{N} R_{k,i_k} > 0 \tag{3}$$

Step1: Let M be the permutation matrix corresponding to $(i_1,\dots,i_N)$ and $C=min(R_{k,i_k})$ for $1\le k\le N$. Construct a new matrix R_s by: $R_s = R-CM$

Step 2: If $C< 1$, the matrix $R_s/(1- C)$ is doubly stochastic, we can find a new permutation of $(i_1,\dots, i_N)$ satisfying (3) and repeat step 1. If $C= 1$, the representation is completed.

C. The scheduling algorithm: algorithm 3 in [7]

Assign a class of tokens for each permutation matrix M_k, $k=1,\dots,K$.

Step1: First, a token is generated for each class. The virtual finishing time of the first class k token is: $V_k^1 = \frac{1}{C_k}$, $k=1,\dots,K$.

Step 2: The switch serves the current class of tokens with the smallest virtual finishing time first.

Step 3: Once the K tokens are served, the next class of K tokens are generated by:

$$V_k^i = V_k^{i-1} + \frac{1}{C_k}, \ k=1,\dots,K;\ i\ge 2.$$

Repeat step 2 and step 3 until all permutation connections have been served with connection duration proportional to their coefficients.

For example, consider the following rate matrix:

$$\lambda = \begin{bmatrix} 0 & 0.3 & 0.2 & 0.4 \\ 0.2 & 0.3 & 0 & 0.2 \\ 0.4 & 0.1 & 0.3 & 0 \\ 0.2 & 0 & 0.2 & 0.3 \end{bmatrix}, \tag{4}$$

where each row represents an input port and each column represents an output port.

Using algorithm 1, we may obtain the following doubly stochastic matrix:

$$R = \begin{bmatrix} 0 & 0.4 & 0.2 & 0.4 \\ 0.4 & 0.4 & 0 & 0.2 \\ 0.4 & 0.2 & 0.4 & 0 \\ 0.2 & 0 & 0.4 & 0.4 \end{bmatrix}, \tag{5}$$

By algorithm 2, we obtain the following decomposition:

$$R = 0.4\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + 0.4\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} + 0.2\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

then, we can set the connection of the switch according to the permutation matrices we obtained above with connection duration proportional to the relative coefficients.

III. QUEUE LENGTH PROPORTIONAL ASSIGNMENT ALGORITHM (QLP)

The integrated algorithm QLP is introduced in this section along with its mathematics analysis.

A. Problem

Let λ be the rate matrix of the guaranteed traffic, and the derived doubly stochastic matrix R be the assigned rate matrix by the scheduling algorithm for both the guaranteed traffic and the best-effort traffic. $R\ge\lambda$ implies that the rate demand of the guaranteed traffic is satisfied, and $R-\lambda$ is the bandwidth assigned to the best-effort traffic. The actual traffic rate matrix B, which can be estimated on line, is the aggregated rate of real guaranteed traffic plus best-effort traffic.

If rate matrix B satisfies the non-overbooking condition (1) and (2), we can simply follow the three algorithms introduced in the previous section by deriving the assigned rate matrix R directly from B, and no further steps are needed. Unfortunately, since there is no admission control for the best-effort traffic, B may fail to satisfy (1) and (2).

Since algorithm 1 derives the assigned rate matrix R from the rate matrix of guaranteed traffic λ only, it may not be able to achieve a high throughput as well as to allocate the leftover bandwidth fairly and efficiently. For example: from (4), the rate demand of the guaranteed traffic from input port 2 to output port 1 is $r_{2,1}=0.2$, but the assigned rate is 0.4 as shown in (5); on the other hand, the rate demand of the guaranteed traffic from input port 3 to output port 1 is $r_{3,1}=0.4$, and the assigned rate is 0.4 too. In other words, a rate of 0.2 is assigned to the best-effort traffic from input 2 to output 1, and none to that from input 3 to output 1. It is possible that the best-effort traffic from input 2 to output 1 may not need all the 0.2 bandwidth, and thus the assigned bandwidth is wasted. On the other hand, the best-effort traffic from input 3 to output 1 cannot be transmitted because no bandwidth is assigned for it.

Chang *et al.* suggested a solution, referred here as the Max-Min algorithm, for this problem in [14] by applying the Max-Min fairness criterion to allocate the bandwidth. The Max-Min fairness originally proposed for flow control [15] is a rate based, light traffic prioritized criterion [16,17,18]. The basic idea is to try to allocate as much network resource as possible to the connection that has the minimum requirement among all connections. The Max-Min fairness can be reached by the "filling procedure." Rate allocation for all input-output pairs increases linearly until the minimum one reaches its rate limitation. Other pairs continue to increase their rates similarly until all bandwidths are allocated. The rate limitation of each pair, which is an element of the actual rate matrix B, can be estimated on line [7]. Since the Max-Min algorithm derives the assigned rate matrix R from not only the rate matrix of the guaranteed traffic λ but also from the estimated rate matrix B, this algorithm can obtain a high throughput by avoiding possible mismatch between the assigned bandwidth and the real traffic load. However, under overloaded conditions, this method can incur a high cell loss ratio if the buffer space is not large enough. Considering a 2 by 2 switch, the available bandwidth of output 1 for the best-effort traffic is 0.5, and actual rates of the best-effort traffic on the two inputs which is destined for output 1 are $r_{1,1}= 0.5$ and $r_{2,1}=0.2$, respectively. To achieve Max-Min fairness, the assigned rate for these two inputs should be: $r_{1,1}= 0.3$ and $r_{2,1}=0.2$. The traffic that cannot be transmitted for the two input ports is $0.2T$ and none, respectively. T is the time interval of the bandwidth allocation procedure. If the condition persists for a long time, the buffer for input 1 is likely to overflow, and the buffer of input 2 is surely under utilized. Another possible problem is the on-line measurement errors may influence the performance of the Max-Min algorithm.

B. QLP for a single output

To achieve a high throughput as well as to improve the utilization of the buffer space, we propose the Queue Length Proportional (QLP) Assignment algorithm to avoid the possible cell loss caused by the Max-Min fairness algorithm under the heavy congestion condition.

The QLP algorithm assigns an input port with a rate proportional to its buffer queue length. The matrix R may be obtained through the following optimization problem.

Maximize:

$$\sum_{i=1}^{N} R_{i,j} \qquad j = 1,\dots,N$$

Subject to:

$$\sum_{i=1}^{N} R_{i,j} \le 1 \quad and \quad \sum_{j=1}^{N} R_{i,j} \le 1 \tag{6}$$

For an N by N switch, let the buffer length of input n $(n=1,\dots,N)$, which is destined to the same output m, $m\in (1,\dots,N)$, be $L_1, L_2,\dots, L_N$, respectively. $L_T= L_1+ L_2+\dots+ L_N$ be the total virtual queue length for output m.

The available bandwidth of output m for the best-effort traffic is R_T. Let $R_1, R_2,\dots,R_N$ be the assigned bandwidth by output m for input 1, 2, …, N, respectively. $R_T=R_1+ R_2+\dots+ R_N$.

Definition 1: *If the allocated rates satisfy the following equation:*

$$\frac{R_i}{L_i} = \frac{R_T}{L_T} = \frac{1}{\mu}, \quad i = 1,2,\dots,N \tag{7}$$

(if $L_i=0$, we set $R_i=0$)
then the allocated rates are called the QLP rates. Otherwise, they are called non-QLP rates.

Definition 2: *Equation (7) is called the proportional rule, and μ the time factor.*

First, we discuss the QLP algorithm for a single output port. Considering the above example: let the current queue lengths for the two inputs be 500 and 200 cells, respectively. Following the proportional rule, the assigned rates are: $r_{1,1}= 0.36$ and $r_{2,1}=0.14$. The traffic that cannot be transmitted after 1000 cell slots for input 1 and 2 are 140 and 60 cells, respectively. If we use the Max-Min algorithm, the traffic that cannot be transmitted for input 1 and 2 are 200 cells and none, respectively. Thus, by using the QLP algorithm, the traffic that cannot be transmitted is balanced between input 1 and 2 to avoid overflow of the buffer at input 1.

It is very interesting to note that: although QLP does not specify any explicit rules for buffer management, we can find from the above example above that QLP inclusively completes the function of buffer sharing. By using QLP, it seems that the input port with heavy traffic load can *steal* the buffer space from the ones with light traffic load by transmitting more cells from its port and delaying the transmission of cells from other ports. It is also one of the reasons why QLP can have lower cell loss ratio than a non-QLP one given a limited buffer space.

Theorem: *The policy for bandwidth assignment for an output port that follows the proportional rule maximize the throughput of best effort traffic.*

Proof:

Let $\underline{L}=[L_1,L_2,\dots,L_N]^T$ be the set of virtual buffer lengths of input ports destined for a same output port.

$\underline{R}=[R_1, R_2,\dots, R_N]$ be the set of the QLP rates.

$\underline{R}(i, j, \varepsilon)=[R_1,\dots,(R_i+\varepsilon),\dots,(R_j-\varepsilon),\dots, R_N]$ be the set of non-QLP rates with a mismatch rate ε happened at input i and j, respectively, where $i,j=1,\dots,N$, $0\le\varepsilon\le R_j$.

Let $T\ge 1$ be the current time interval for bandwidth allocation procedure. If $T=1$, the bandwidth allocation is performed per time slot. Also, let β be the allowed transmission time of the best-effort traffic, and $T-\beta$ be the transmission time of the guaranteed traffic. During each time interval T, the total best-effort traffic transmitted by using QLP and non-QLP is S_Q and S_{NQ}, respectively:

Case 1: $\beta<\mu$, in this case, not enough bandwidth is available for best-effort traffic.

$$S_Q = R_T\beta$$

$$S_{NQ} = (R_T - R_i - R_j)\beta + \min(L_i,(R_i+\varepsilon)\beta) + (R_j-\varepsilon)\beta$$

$$\text{If } L_i<(R_i+\varepsilon)\beta \Rightarrow \varepsilon > L_i/\beta - R_i \tag{8}$$

$$S_Q > S_{NQ}$$

Otherwise: $S_Q = S_{NQ}$

Case 2: $\beta=\mu$, in this case, there is an exact bandwidth for the best-effort traffic.

$$S_Q = R_T\beta = L_T$$

$$S_{NQ} = (R_T - R_i - R_j)\beta + \min(L_i,(R_i+\varepsilon)\beta) + \min(L_j,(R_j-\varepsilon)\beta)$$
$$= (R_T - R_i - R_j)\beta + R_i\beta + (R_j-\varepsilon)\beta < S_Q$$

Case 3: $\beta>\mu$, in this case, there is more than enough bandwidth for all best-effort traffic.

$$S_Q = R_T\mu = L_T$$

$$S_{NQ} = (L_T - L_i - L_j) + L_i + \min(L_j,(R_j-\varepsilon)\beta)$$

$$\text{If } L_j > (R_j-\varepsilon)\beta \Rightarrow \varepsilon > R_j - L_j/\beta \tag{9}$$

$$S_Q > S_{NQ}$$

Otherwise: $S_Q = S_{NQ}$

Thus, a non-QLP algorithm cannot transmit more traffic than a QLP one. ■

***Corollary 1**: The further time delay caused by a non-QPL algorithm compared to the QPL one is decided by the mismatch rate ε, the time factor μ, transmission time of best-effort traffic β, and related queue lengths L_i or L_j.*

Proof:

Let δ be the further time delay caused by a non-QLP algorithm. Consider the same cases as in the above theorem.

Case1:

$$\delta = ((R_i+\varepsilon)\beta - L_i)/(R_i+\varepsilon) = \beta - \mu/(1+\mu\,\varepsilon/L_i) \quad (10)$$

subject to (8).

Case 2:

$$\delta = \varepsilon\beta/(R_j-\varepsilon) = \beta/(L_j/(\mu\,\varepsilon)-1) \quad (11)$$

Case 3:

$$\delta = (L_j-(R_j-\varepsilon)\beta)/(R_j-\varepsilon) = \mu/(1-\mu\,\varepsilon/L_j) - \beta \quad (12)$$

subject to (9).

From (10)-(12), we can conclude that the further time delay is decided by ε, β, μ, L_i or L_j. ■

C. QLP for a switch

Although QLP maximizes the throughput of best effort traffic, unfortunately, the QLP rates may not always be approached for a switch that has multiple output ports limited by condition (6). Thus, we need to find the working area of the optimal bandwidth assignment in term of throughput for a switch. Fig. 1 and Fig. 2 show the rate assignment for a 2 by 2 switch under the condition of $\beta \le \mu$ and $\beta > \mu$, respectively, where the x- and y-axis represent the rates assigned to input port i and j, respectively.

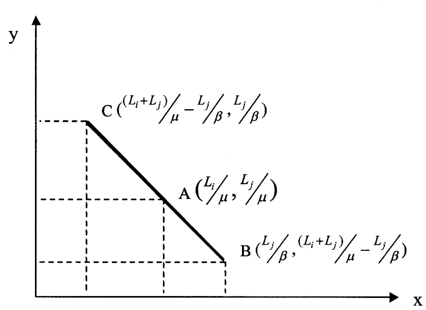

Fig.1. Rate assignment that maximizes best-effort traffic throughput for $\beta \le \mu$

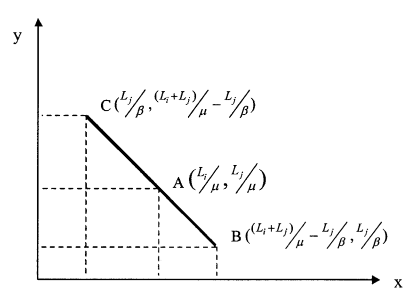

Fig.2. Rate assignment that maximizes best-effort traffic throughput for $\beta \ge \mu$

As shown in Fig. 1 and Fig. 2, the rates that maximize throughput take on values on line AB (or AC). Line AB implies a mismatch rate of $\varepsilon \ge 0$ is added to R_i and subtracted from R_j; Line AC, on the other hand, implies that ε is added to R_j and subtracted from R_i. Point A represents the proportional rates. Although the assigned rates taken on point B (or C) can also obtain a maximum throughput, they cannot approach a high utilization of buffers, e.g., taking rate values on point B, the buffer for input j may be full, and the buffer for input i will be under utilization.

D. Fairness of QLP

QLP follows the Queue Proportional Fairness (QPF) criterion instead of the Max-Min fairness. QPF criterion, which employs the cell loss ratio as the fairness metric, is proposed to efficiently allocate both buffer space and bandwidth to the best effort traffic.

Although a buffer management scheme such as POT can prevent misbehaving users from hogging the whole buffer space at each input port, we still need to limit overloaded users from occupying too much bandwidth from users in other input ports by setting a maximum length threshold L_M. If $L_i > L_M$, we set $L_i = L_M$ in equation (7), $i=1,\ldots,N$.

IV. SIMULATIONS

We use a 4×4 non-blocking crossbar switch to evaluate the performance of QLP and the Max-Min algorithm. We choose POT as the buffer management policy to manage the buffer at each input port for both cases. We assume that all the input ports and output ports have the same transmission rates, and we normalize the rates of input ports by dividing them by that of the output ports.

The traffic is generated at each input port as a full-loaded one with ρ=1. To evaluate both algorithms in the severe overloaded condition, we simply assume that all traffic from input 1 goes to output 1; 50% of traffic from input 2 goes to output 1, and another 50% goes to the output 2; 80% of traffic from input 3 goes to output 3, and the other 20% goes to output 4; half of the traffic from input 4 goes to output 3 and another half goes to output 4. Note that there is not enough bandwidth for output 1 and 3.

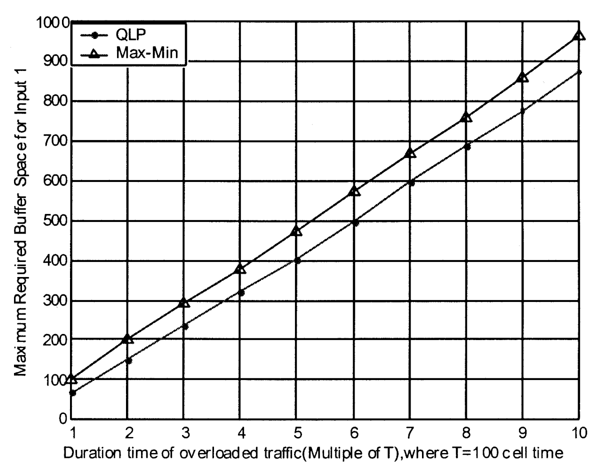

Fig.3. Comparison of maximum required buffer space using QLP and the Max-Min algorithm for input 1.

Fig. 3 shows the required buffer space for no cell loss in input port 1 using the Max-Min algorithm and the QLP algorithm, respectively. It is shown that, with the same traffic condition, the switch requires less buffer space at each input port by using the QLP algorithm than that by the Max-Min algorithm.

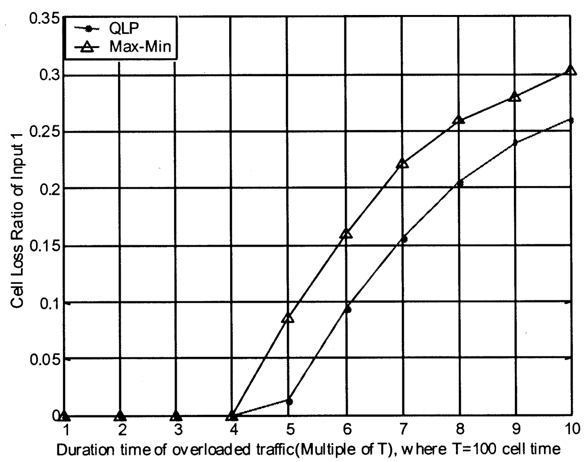

Fig.4. Cell loss ratio of input port 1 using QLP and the Max-Min algorithm.

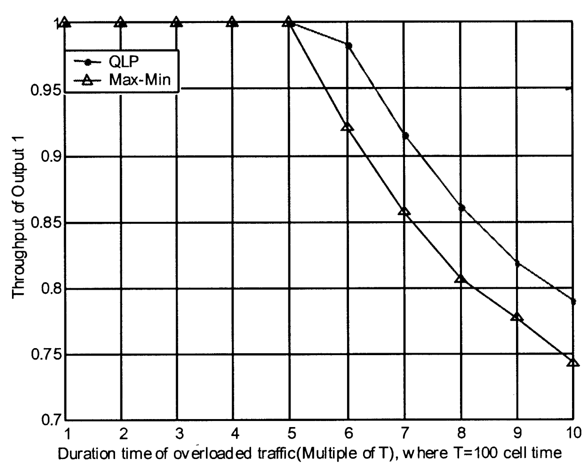

Fig.5. Throughput of output port 1 using QLP and the Max-Min algorithm.

As shown in Fig. 4, if the buffer space for each input port is limited to 400 cells, the cell loss ratio using the QLP algorithm is around 25% to 100% lower than that using the Max-Min algorithm.

The throughput of QLP as shown in Fig. 5 has improved by about 6% as compared to that of the Max-Min algorithm.

V. CONCLUSION

A joint scheduling and buffer management algorithm QLP for input queued switches is presented in this paper. The allocation of bandwidth is based on the real traffic queue length as well as the available bandwidth so that neither buffer space nor bandwidth will be wasted for possible mismatch between them. Since QLP considers constrain of the buffer space inside a switch, it can achieve not only a high throughput but also a low cell loss ratio.

Another salient feature for QLP is that the heavy load traffic in an input port can logically share buffers of other input ports although there are no physical connections among them.

QLP is most suitable for handling congestion caused by the bursty traffic, hot-spot traffic, and malicious users.

Acknowledgements: This work was supported in part by the NJ Commission on Science and Technology via the NJ Center for Wireless Telecommunications, and the NJ Commission on Higher Education via the NJI-TOWER project.

REFERENCES

1. F. Tobagi, "Fast Packet Switch Architectures for Broad-band Integrated Services Digital Networks." Proc. Of the IEEE, Col.78, No. 1 Jan. 1990, pp133-167.
2. K. Kou, "Realization of Large-capacity ATM Switches," IEEE communication magazine. Dec. 1999, pp120-123.
3. G. Wu and J. Mark, " A Buffer Allocation Scheme for ATM Networks: Complete Sharing Based on Virtual Partition," IEEE/ACM TRANSACTION ON NETWORKING, Vol . 3, No.6, Dec 1995. pp660-670.
4. Cidon, L. Georgiadis and R. Guerin, " Optimal Buffer Sharing," in Proc. IEEE INFOCOMM'95, pp. 24-31, 1995.
5. S. Sharma and Y. Viniotis, "Optimal Buffer Management policies for Shared-Buffer ATM Switches," IEEE/ACM TRANSACTION ON NETWORKING, Vol.7, No. 4, Aug. 1999, pp575-587.
6. N. McKeown, V. Anantharam and J. Walrand, "Achieving 100% throughput in an input-queued switch," in Proc. IEEE INFOCOM'96, pp.296-302,1996.
7. C. Chang, W. Chen and H. Huang, " On Service Guarantees for Input Buffered Crossbar Switches: A Capacity decomposition approach by Birkhoff-von Neumann," IEEE IWQoS'99, pp.79-86, London, U.K, 1999
8. N. McKeown and A. Mekkittikul, "A Practical Scheduling algorithm to achieve 100% throughput in input-queued switches," in Proc. IEEE INFOCOM'98, pp.792-799,1998.
9. N. McKeown and A. Mekkittikul, "A Starvation free algorithm for achieving 100% throughput in an input-queued switch," in Proc. IEEE ICCCN'96, pp.226-229,1996.
10. N. McKeown, "The iSLIP Scheduling Algorithm for Input-Queued Switches," IEEE Transaction on Networking, Vol.7, No.2, April 1999. pp180-201.
11. M. Marsan, A. Bianco, E. Leonardi and L. Milia, "RPA: A Flexible Scheduling Algorithm for Input Buffered Switches," IEEE Transaction on Communications, Vol.47, No.12, December 1999. pp1921-1933.
12. G. Lapiotis, " Stochastic Analysis of Joint Buffer Management and Service Scheduling in High-Speed Network Nodes", Ph. D Thesis, Polytechnic University, 1999.
13. R. Guerin, S. Kamat, V. Peris and R. Rajan, "Scalable QoS Provision Through Buffer Management," in Proc. IEEE SIGCOMM'98, pp.29-40. 1998.
14. C. Chang, W. Chen and H. Huang, " Birkhoff-von Neumann Input Buffered Crossbar Switches," in Proc. IEEE INFOCOM'00, pp.1614-1623, 2000.
15. D. Bersekas and R. Gallager, Data Networks, Prentice Hall, 1992.
16. Kam and K. Siu, " Linear Complexity Algorithm for QoS Support in Input-Queued Switches with No Speedup," IEEE JSAC, Vol.17, No.6, 1999, pp1040-1056.
17. F. Kelly, "Charging and Rate Control for Elastic Traffic", *European Trans. On Telecomm.* Vol.8 pp33-37,1997.
18. L. Massoulie and J. Roberts, "Bandwidth Sharing: Objectives and Algorithms", in Proc. Infocom'99, pp1395-1403,1999.