

## Research Article

# DQOBSR Protocol with QoS Provisioning for OBS Metro Rings

M. Thachayani<sup>1</sup> and R. Nakkeeran<sup>2</sup>

<sup>1</sup> Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry, India

<sup>2</sup> Department of Electronics Engineering, School of Engineering and Technology, Pondicherry University, Puducherry, India

Correspondence should be addressed to M. Thachayani; [mthachayani@pec.edu](mailto:mthachayani@pec.edu)

Received 28 April 2013; Revised 10 August 2013; Accepted 11 August 2013

Academic Editor: Haroldo Hattori

Copyright © 2013 M. Thachayani and R. Nakkeeran. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Optical burst switching (OBS) is the switching technology that enables optical internet with current technology. OBS networks are capable of operating with or without optical buffers and wavelength converters. Hence, medium access control (MAC) protocol plays an important role in resolving contention in such networks. Distributed Queue OBS Ring (DQOBSR) is one MAC protocol proposed for OBS ring networks. In this paper, the quality of service (QoS) provisioning scheme for DQOBSR protocol is presented. For the proposed protocol, service differentiation for different classes of traffic in terms of throughput and access delay is verified through simulation. A mathematical model is also presented for the computation of classwise mean access delay. The results obtained using the model are found to be in good agreement with the simulation results.

## 1. Introduction

OBS is seen as the prospective solution for optical internet that enables efficient utilization of the fiber, withstanding the current technological limitations [1]. In OBS, contention resolution is one of the most important issues, since the network is provided with limited or no optical buffers. An effective method of addressing the contention is selecting an appropriate MAC protocol. A new MAC protocol, namely, distributed access OBS (DAOBS) is proposed for OBS rings in [2], and later it had been extended for mesh topologies [3]. This protocol is an adaptation of distributed queue dual bus (DQDB) to OBS ring networks. For the case of ring topology, DQOBSR protocol [4] is proposed with simpler bus assignment without compromising the performance. In this paper, the QoS provisioning for DQOBSR-based networks is presented, and the performance is analyzed through simulation and mathematical modeling.

## 2. Overview of DQOBSR Protocol

The topology for the DQOBSR protocol consists of two uni-directional counter rotating rings similar to fiber distributed data interface (FDDI). The capacity of each ring is split into

$W$  data channels and two control channels. The schematic diagram of the network implementing the DQOBSR MAC protocol is shown in Figure 1. Each wavelength channel is divided into equal length slots. The control slots are further split into  $W$  number of *minislots*, and there is a one-to-one correspondence between the *minislots* and the  $W$  number of data channels. The control slot format is shown in Table 1.

The nodes implement a first-in-first-out (FIFO) distributed queue similar to the one used with DQDB protocol using the two counters, namely, the *request counter* (RC) and the *countdown counter* (CDC). However there are two major differences: single node acts as *headend* node for both the rings in DQOBSR protocol and the bus is wavelength multiplexed. The multiplexed bus necessitates an entirely different control scheme and slot format. The function of the *headend* is to generate empty slots for the control channels, that is, slots with starting and ending delimiters. With this, clock recovery can be done all optically using simple optical processing devices [5]. The nodes are equipped with a tunable transmitter and tunable receiver for sending and receiving the bursts.

Before selecting the slot for transmission, the node ensures that no other burst is destined for the same node, as the destination of the burst to be transmitted is in the same

TABLE 1: Control slot format.

Preamble	Busy/free	Destination-id	Source-id	Class	Future use
3 bits	1 bit	8 bits	8 bits	2 bits	1 bit

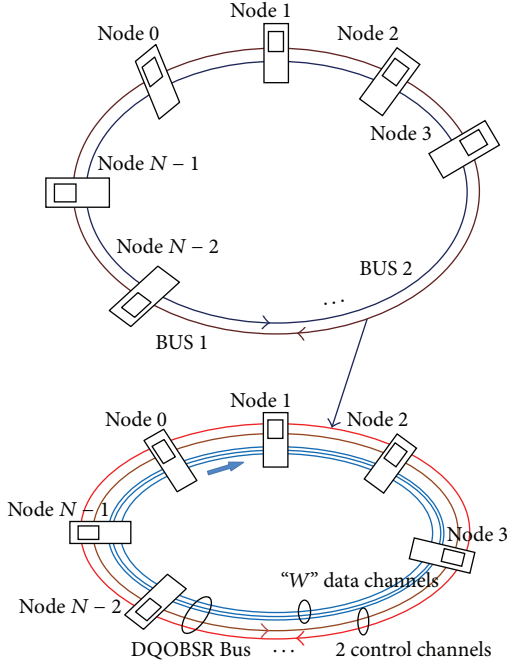


FIGURE 1: Schematic diagram of the network implementing DQOBSR protocol.

slot. Then the burst is transmitted using the corresponding wavelength based on the position of the *minislot* after an offset time similar to conventional OBS [1]. However, when there are no requests from the downstream nodes, then any node can make use of the passing free slot without making a reservation.

### 3. QoS Provisioning of DQOBSR Protocol

Since the protocol is based on distributed queuing, QoS provisioning can be implemented through priority queuing. In this priority scheme, at any point in time, a reservation request is sent for the lower priority bursts only if the higher priority queue is empty at the nodes. Moreover, an upstream node receiving a request for a lower priority burst will consider the request and, allow a free slot only if it does not have any higher priority bursts to transmit at that time. However, within a particular priority class the service is according to FIFO discipline.

When there are  $M$  priority classes, each node will consist of  $M$  number of RC and CDC pairs, each  $RC_j$  and  $CDC_j$  pair associated with the corresponding queue for the  $j$ th priority level. At a particular priority level, a node can be in *IDLE* or *BUSY* state. A node is said to be in *IDLE* state when there are no bursts waiting for transmission. In this state, the node will

be updating the  $RC_j$  based on the requests from downstream nodes and higher priority self-burst arrivals. The node is in *BUSY* state when it has a burst to transmit at that priority level. In this state, after placing a request, it will update the  $CDC_j$  based on the requests from downstream nodes and higher priority self-burst arrivals. A higher priority burst is given priority at both local as well as at distributed queues by the following actions.

A node in *IDLE* state at the  $j$ th priority level monitors the requests from downstream nodes as well as bursts arriving to the node itself, with a priority level equal to or higher than  $j$ . For each higher priority request or burst arrival, the node increments the request counter,  $RC_j$ . Similarly, a node in *BUSY* state increments the countdown counter,  $CDC_j$  for every high-priority self-burst arrival and every high priority request arrival.

### 4. Mathematical Model

Let the delay incurred in the local queue be  $W_{lq}$  and let the delay incurred in the distributed queue be  $W_{dq}$ . Then the total queuing delay,  $T_q$  can be expressed as

$$T_q = W_{lq} + W_{dq}. \quad (1)$$

The total number of priority classes is  $M$ . The per slot arrivals can be characterized by the joint probability generating function as follows:

$$\Gamma(z) \triangleq E \left[ \prod_{j=1}^M z_j^{\lambda_{jk}} \right]. \quad (2)$$

Similarly, the common probability mass function of identical and independently distributed service times can be represented by  $S_j(z) \triangleq \sum_{m=1}^{\infty} s_{j(m)} z^m$ , and the mean service time of a *class-j* burst is given by  $t_j = S'_j(1)$ . Then the load offered by *class-j* bursts is given by  $\rho_j \triangleq \lambda_j t_j$ , and the total load is given by  $\rho_T \triangleq \sum_{j=1}^M \rho_j$ . *Class-i* bursts are assumed to have nonpreemptive priority over *class-j*, if  $i < j$ . Using the approach in [6] for the head of line (HoL) priority scheme, the mean delay for *class-j* bursts in the distributed queue can be expressed as

$$W_{dqj} = t_{dj} + \frac{\lambda_{jj} t_{dj}}{2\lambda_j (1 - \sum_{m=1}^j \rho_m)} + \frac{\sum_{m=1}^j \lambda_{mj} t_{dm}}{\lambda_j (1 - \sum_{m=1}^j \rho_m)} + \frac{\sum_{m,n=1}^{j-1} \lambda_{mn} t_{dm} t_{dn}}{2(1 - \sum_{m=1}^{j-1} \rho_m)(1 - \sum_{m=1}^j \rho_m)} + \frac{\sum_{m=1}^M \lambda_m t_{mm}}{2(1 - \sum_{m=1}^{j-1} \rho_m)(1 - \sum_{m=1}^j \rho_m)}, \quad (3)$$

where

$$\lambda_{ij} \triangleq \frac{\partial^2 \Gamma(z_1, \dots, z_M)}{\partial z_i \partial z_j} \Big|_{z_1 = \dots = z_M = 1}, \quad (4)$$

$$t_{jj} \triangleq \frac{d^2 S_j(z)}{dz^2} \Big|_{z=1}.$$

The Poisson arrivals and deterministic service times are chosen for calculating the delay for DQOBSR-based network.

The priority-based local queuing delay,  $W_{lqj}$ , can be written as

$$W_{lqj} = t_{ij} + \frac{\lambda_{jj} t_{ij}}{2\lambda_j (1 - \sum_{m=1}^j \rho_m)} + \frac{\sum_{m=1}^j \lambda_{mj} t_{lm}}{\lambda_j (1 - \sum_{m=1}^j \rho_m)}$$

$$+ \frac{\sum_{m,n=1}^{j-1} \lambda_{mn} t_{lm} t_{ln}}{2(1 - \sum_{m=1}^{j-1} \rho_m)(1 - \sum_{m=1}^j \rho_m)} \quad (5)$$

$$+ \frac{\sum_{m=1}^M \lambda_m t_{mm}}{2(1 - \sum_{m=1}^{j-1} \rho_m)(1 - \sum_{m=1}^j \rho_m)},$$

where

$$t_{lqj} = W_{daj} + t_{off} + t_t, \quad (6)$$

$t_{lqj}$  = holding time in the local queue for class- $j$  bursts, and  $t_{off}$  = offset time,  $t_t$  = transmission time (1 slot).

Then the total access delay can be obtained by substituting (3) and (5) in (1).

## 5. Simulation Results

The performance parameters analyzed are mean access delay, blocking probability, and throughput. The bursts are generated according to Poisson process. The burst size is fixed to be one slot size which is set to 0.1 msec. The capacity of each channel is taken as 10 Gbps. The processing time is assumed to be 10  $\mu$ s. The length of each link is assumed to be 50 Kms. Three service classes are used for the bursts, *class-0*, *class-1*, and *class-2*. The simulation results presented for a 16 node network with a typical traffic proportion of 0.1 (*class-0*), 0.2 (*class-1*), and 0.7 (*class-2*). The *class-0* and *class-1* refer to the delay sensitive higher priority classes, and *class-2* corresponds to the best-effort traffic class. *Class-0* corresponds to traffic with very strict delay constraint like interactive audio/video and *class-1* corresponds to relatively tolerant but delay sensitive traffic such as audio/video streaming. The discrete event simulator is written in Java and run using the Eclipse IDE.

**5.1. Throughput.** The normalized throughput is determined by taking the ratio between the total number of bursts transmitted and the total number of bursts generated. Figure 2 shows the throughput performance of different classes. Throughput decreases with the increase in load irrespective of the service class. At loads less than 0.5 Erlangs, there is no significant difference in throughput between the classes. However, for higher loads of more than 0.5 Erlangs,

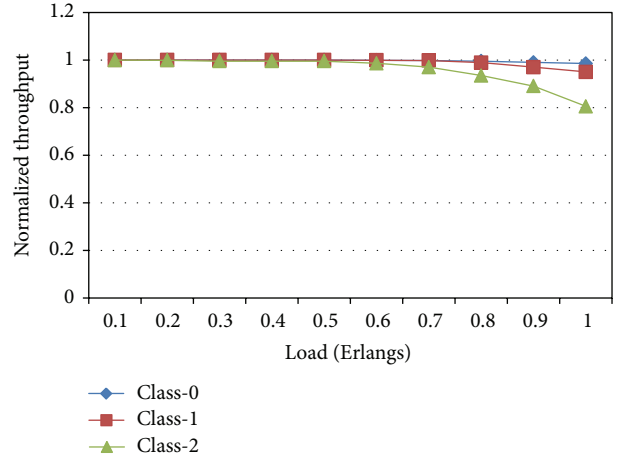


FIGURE 2: Throughput comparison of the two different classes.

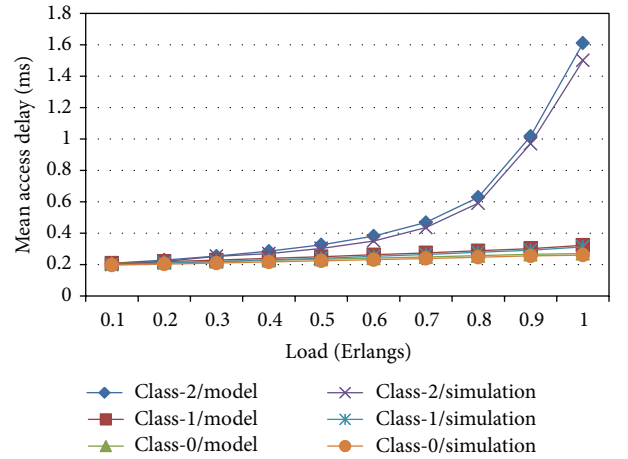


FIGURE 3: Comparison of mean access delay between classes obtained through simulation and model.

the service difference between classes is observed. There is a difference of more than 3% and 18% observed in the throughput of *class-1* and *class-2* bursts, respectively, with respect to *class-0* bursts at 1 Erlangs. Because of the prioritized scheduling at the local and distributed queues, the probability of a *class-0* burst getting transmitted is more compared to the lower priority bursts.

**5.2. Mean Access Delay.** Mean access delay is the interval between the instant a burst arrives at a node to the instant it gets a free slot and starts transmission. The mean access delay performance for the three classes obtained through simulation and model is compared in Figure 3. The delay of *class-0* and *class-1* bursts is less by almost 83% and 80%, respectively, when compared with *class-2* bursts.

The delay values obtained from the model is found to be in good agreement with the values obtained using simulation. Mean value of the error is found to be 5.1% of the mean delay value. The deviation may be attributed to the single server

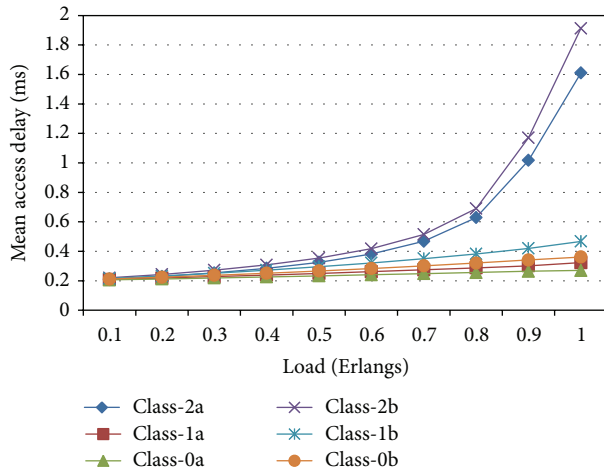


FIGURE 4: Comparison of mean access delay between classes for different relative proportions.

model approximation of the wavelength multiplexed ring of DQOBSR-based network.

If the relative proportion of the higher priority traffic is increased, the access delay will also get increased. The mean access delay for the three different classes is plotted for two different traffic combinations in Figure 4. *Case a* refers to the 0.1/0.2/0.7 (*class-0/class-1/class-2*) distribution of traffic chosen for all other plots, and *case b* refers to relative proportion of 0.2/0.3/0.5 traffic. There is an increase in delay observed for all the traffic classes in *case b* with respect to the corresponding delay in *case a*. In case of *class-1*, there is an increase of 23.7% of delay with respect to *case a*. This is due to the relative increase in the higher priority *class-0* and increase in its own *class-1* traffic as well. The mean access delay of *class-0* traffic is also increased by 17.7% due to the increase in its own *class-0* traffic from 0.1 to 0.2%. For the case of lowest priority *class-2* traffic, the increase in delay is 12.8%. There is an increase of its own priority level traffic from 50% to 70%, which should result in increased delay for *class-2* traffic for *case b*. However, the reduction of 20% in the higher priority traffic has greater impact in the access delay of the lowest priority class leading to reduced delay. Hence, the relative increase in delay for *class-2* is between the *class-0* and *class-1* traffic classes.

From Figures 2, 3, and 4, it can be observed that effective QoS provisioning can be implemented with the proposed priority-queueing-based scheme for DQOBSR networks.

**5.3. Blocking Probability.** Blocking probability is determined by taking the ratio between number of bursts dropped due to unavailable buffer space and total number of bursts generated. With sufficient electronic buffer capacity at the nodes, it is possible to achieve zero blocking at the expense of increased access delay. The blocking probability performance for the three classes is shown in Figure 5. For buffer capacities of above 5000 bursts, no blocking is observed. For a buffer size of 1000 bursts, no blocking occurs for the higher priority classes but blocking is observed for *class-2* bursts. For a

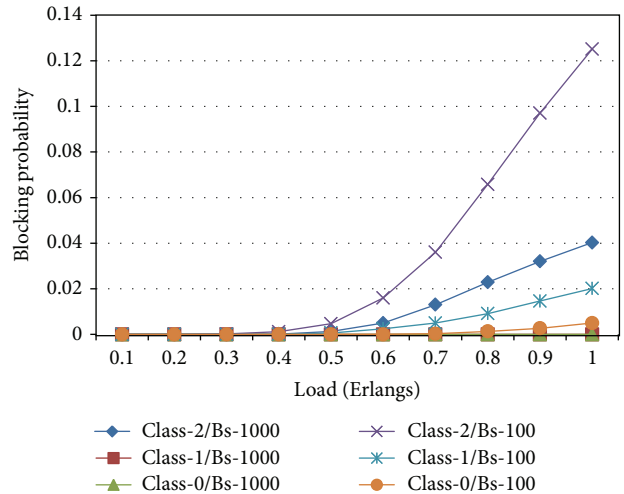


FIGURE 5: Comparison of blocking probability between different classes.

reduced buffer size of 100, blocking occurs irrespective of the class of service.

It can also be observed that blocking is present for the *class-2* bursts even at lower loads (<0.5 Erlangs), whereas for *class-0* bursts, blocking begins to occur at relatively larger loads (>0.7 Erlangs) even at small buffer sizes. This shows the effectiveness of the QoS provisioning scheme.

## 6. Conclusion

In this paper the QoS provisioning through priority assignment is presented and analyzed for the distributed-queue-based DQOBSR protocol. From the simulation results, it is observed that the proposed priority assignment in the local and distributed queues is able to provide very good service differentiation among classes. A mathematical model is also presented to compute the classwise access delay and the predicted values are found to be in good agreement with the simulation results. Though the scheme is shown for only three service classes, the concept can be extended for more number of classes, if required.

## References

- [1] M. Yoo and C. Qiao, "Optical burst switching (OBS): a new paradigm for an Optical Internet," *Journal of High Speed Networks*, vol. 8, no. 1, pp. 69–84, 1999.
- [2] J. Triay and C. Cervelló-Pastor, "Distributed contention avoidance in optical burst-switched ring networks," in *Proceedings of the 11th IEEE Singapore International Conference on Communication Systems (ICCS '08)*, pp. 715–720, Guangzhou, China, November 2008.
- [3] J. Triay, G. S. Zervas, C. Cervelló-Pastor, and D. Simeonidou, "Multiservice QoS-enabled MAC for optical burst switching," *Journal of Optical Communications and Networking*, vol. 2, no. 8, Article ID 5520648, pp. 530–544, 2010.
- [4] M. Thachayani and R. Nakkeeran, "WDMA protocol for OBS ring metro network with distributed queueing," in *Proceedings*

*of the International Conference on Process Automation, Control and Computing (PACC '11)*, pp. 1–4, July 2011.

- [5] E. Kehayas, K. Vysokinos, L. Stampoulidis, K. Christodoulou, K. Vlachos, and H. Avramopoulos, “ARTEMIS: 40-Gb/s all-optical self-routing node and network architecture employing asynchronous bit and packet-level optical signal processing,” *Journal of Lightwave Technology*, vol. 24, no. 8, pp. 2967–2977, 2006.
- [6] J. Walraevens, B. Steyaert, M. Moeneclaey, and H. Bruneel, “Delay analysis of a HOL priority queue,” *Telecommunication Systems*, vol. 30, no. 1–3, pp. 81–98, 2005.

