Journal of Digital Convergence
Vol. 20. No. 3, pp. 389-397, 2022

ISSN 2713-6434 / eISSN 2713-6442
https://doi.org/10.14400/JDC.2022.20.3.389

Adaptive Burst Size-based Loss Differentiation for Transmitting Massive Medical Data in Optical Internet

Yonggyu Lee
independent researcher

광 인터넷에서 대용량 의학 데이터 전송을 위한 적응형 버스트 길이 기반 손실 차등화 기법

이용규
독립연구자

Abstract As increasing the growth of the Internet in medical area, a new technology to transmit effectively massive medical data is required. In optical internet, all OBS nodes have fiber delay lines, hardware components. These components are calculated under some optimal traffic conditions, and this means that if the conditions change, then the components should be altered. Therefore, in this article a new service differentiation algorithm using the previously installed components is proposed, which is used although the conditions vary. When traffic conditions change, the algorithm dynamically recalculates the threshold value used to decide the length of data bursts. By doing so, irrelevant to changes, the algorithm can maintain the service differentiation between classes without replacing any fiber delay lines. With the algorithm, loss sensitive medical data can be transferred well.

Key Words : Transmission of Medical Data, QoS, OBS, Fiber Delay Line, Dynamic Burst Size

요 약 의학 분야에서 인터넷 활용의 증가로 대용량 의학 데이터를 효율적으로 전송할 수 있는 기술이 요구되고 있다. 광 인터넷에서 모든 OBS 노드들은 하드웨어 컴포넌트인 광 지연 라인들을 가지고 있다. 이것들은 몇 가지 최적 트래픽 조건을 이용해 계산되기 때문에 트랙픽 조건이 변하면 광 지연 라인들도 변해야 한다는 것을 의미한다. 이에 본 논문에서는 트랙픽 조건이 변하더라도 기존에 설치된 광 지연 라인을 이용하는 서비스 차등화 알고리즘을 제안한다. 트래픽 조건이 변할 때, 새로운 알고리즘은 데이터 버스트의 길이를 결정하기 위해서 사용되는 스레쉬홀드 값을 동적으로 계산한다. 그러므로 트래픽 조건이 변할지라도 제안된 알고리즘은 광 지연 라인의 대체 없이도 클래스들 사이에서 서비스 차등화를 달성할 수 있다. 본 알고리즘을 이용하면 손실에 민감한 대용량 의학 데이터를 효율적으로 전송할 수 있다.

주제어 : 의학 데이터 전송, 서비스 품질, OBS, 광 지연 라인, 동적 버스트 크기

---

*Corresponding Author : Yonggyu Lee(lyonggyu@gmail.com)

Received January 21, 2022
Revised February 12, 2022
Accepted March 20, 2022
Published March 28, 2022

1. Introduction

With the advent of a great variety of Internet Protocol-based services and applications, especially real-time multimedia services, increasing demands for transmission bandwidth have given rise to wavelength division multiplexing (WDM) technologies such as OCS (Optical Circuit Switching), OLS (Optical Label Switching), and OPS (Optical Packet Switching).

On the other hand, Optical burst switching (OBS) receiving much considerable attention in the past few years is a hybrid approach of out-of-band signaling while data packets remain in the optical domain all the time [1].

OBS technologies are studied to provide an optical Internet backbone, as it eliminates the electronic bottleneck at intermediate switching nodes and guarantees QoS (Quality-of-Service) without any buffering [1]. In OBS networks, there is a strong separation between the control and data planes, which allows for great network manageability and flexibility. In addition, its dynamic nature leads to high network adaptability and scalability, which makes it quite suitable for transmission of bursty traffic.

However, despite the above advantages, in order to implement practical OBS networks, there are still a lot of challenging problems to be solved. Among many problems, one of the most important issues that must be dealt with is the need for service differentiation schemes for various applications [2-6].

Until now, various QoS mechanisms have been proposed as follows. In [7], a mechanism to guarantee QoS was proposed even when there are some problems such as fiber cut and high loss rate during propagating to the destination. In [8], the authors proposed a new QoS scheme for typical MAN architectures based on OBS ring networks. In [9], a QoS scheme was proposed based on GMPLS-enabled OBS network architectures. In [10], the burst segmentation scheme allows high-priority bursts to preempt low-priority bursts and enables full class isolation between bursts. In [11], an absolute QoS scheme was proposed to support hard QoS in OBS networks. In [12], proportional QoS differentiation is provided by maintaining the number of wavelengths occupied by each class of bursts. In [13, 14], in order to achieve service differentiation. a new algorithm through fiber delay lines was proposed.

In the most previous works including the above references, their schemes used fixed data burst length to accomplish service differentiation. However, in this paper, a new service differentiation algorithm, which changes the length of data burst, is proposed.

This article is organized as follows. Section 2 shows the various buffering architectures for OBS nodes. Section 3 displays our new algorithm. Section 4 illustrates the performance analysis of our proposed algorithm. Finally, Section 5 concludes the article.

2. Buffering Architecture

In this section, buffering architecture to be considered is shown. Figure 1 shows a feed-forward output buffering architecture. In the architecture, each wavelength has a dedicated FDL bank, which consists of many FDLs, and a direct FDL, which has a zero delay value. With sufficiently large output buffers, a feed-forward output buffered switch can achieve the best possible burst loss performance. Sometimes, if the buffer size is arbitrarily large, then burst loss can be zero [15]. However, increasing the buffer size is limited in practice because large buffers will increase the cost of FDLs. Nevertheless, because a switch fabric is assumed as non-blocking in the most cases, the numerical analysis of FDL buffers can be easily made in this architecture [16, 17].

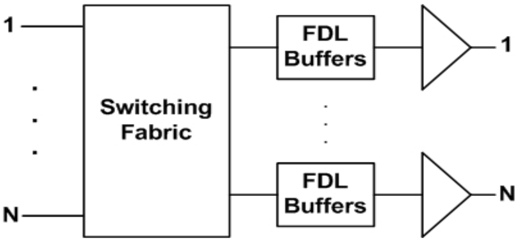


Fig. 1. Feed-forward output buffering architecture.

In addition to the buffering architectures and the number of FDLs, the burst forwarding performance can be affected by two major issues [16]. The first issue is the distribution of the lengths of the FDLs. FDL buffers can be configured as either degenerate buffers (linear increment of delay line lengths) or non-degenerate buffers (non-linear increment of delay line lengths). Degenerate means that the lengths of FDLs are consecutive multiples of a certain delay unit (D) called basic delay unit, but non-degenerate means that the lengths of FDLs can be an arbitrary set that does not require consecutive multiples of a certain delay unit (D).

Another important factor is the packet forwarding policy. Generally, there are two kinds of scheduling policies: void-filling and non-void-filling, where a void is the interval between the departure of the last bit of a packet and the first bit of the next packet at a certain output port of the switch. In OBS networks, such voids may occur because the FDLs can provide only a fixed amount of delay. With a void-filling policy, an incoming packet can be switched to any FDL, or to fill any void, as long as there is no contention at the output port. As a result, the FIFO (First-In-First-Out) discipline of certain output ports may be violated. In some previous works [20, 21], it has been shown that, under certain conditions, a void-filling scheduling policy may be more efficient than a non-void-filling. Nevertheless, a void-filling scheme may lead to packet ordering problems and may require more complex control overhead.

In this paper, Fig. 1 architecture, feed-forward output buffering architecture, is considered. Also, a degenerate FDL length configuration and a non-void-filling scheduling scheme are considered. A FIFO ordering discipline is considered as the scheduling policy for data bursts.

3. Loss Differentiation Algorithm

In our previous work, service differentiation could be accomplished using those similar conditions, a fixed optimal delay unit and a certain traffic load [13]. However, as shown in the work, according as the offered load increases, the lowest class, best effort service, can hardly transmit traffic. Also, the variation of the number of FDLs for each sub-FDL group is too big.

Therefore, in this article, a new service differentiation algorithm will be proposed to solve the aforementioned problems. In [13], it is shown that as the data burst lengths change, the burst loss probabilities are altered. This means that if the data burst lengths are carefully manipulated, the problems of the previous paper can be resolved. In our new scheme, core nodes send some information about traffic conditions to edge nodes, which transmitted contending data bursts to core nodes, to control the average data burst length. Each edge node can control the size of data bursts with those information.

3.1 Algorithm

The original algorithm has a simple function that if a sub-FDL group succeeds transmitting a data burst and the group has enough number of FDLs, the algorithm reduces the number of FDLs of the group by one, but if the group fails sending a data burst and the group does not have enough number of FDLs, the algorithm increases the number of FDLs of the group by one [13]. In other words, when occuring contentions at class k, each core node calculates the current blocking probability of the class, BC_c^k, and BC_c^k is compared with the target blocking probability of the class, BC_t^k. If BC_c^k is greater than BC_t^k, then an over−counter indicator for the class, OC_k, is increased by one. Also, if OC_k is greater than the over−threshold−counter indicator for class k, OT_k, then the number of FDLs for the group, FC_k, is compared with the maximum number of FDLs of the group, $FC_k^{\max}$. If FC_k is less then $FC_k^{\max}$, then FC_k is increased by one and the number of FDLs for best−effort traffic class, FC_n, is decreased by one. Then, OC_k is set to zero, and if the current blocking probability of class n for best−effort service, BC_c^n is greater than the target blocking probability for the class, BC_t^n, then each core node sends some feedback messages to the edge nodes delivering the contending data bursts as shown in Algorithm 1.

When each edge node receives the adjustment messages from the above Algorithm 1, it runs data burst size decision algorithm shown in Algorithm 2 to change the length of data bursts. When each edge node receives the message, it increases an over−indicator counter, OI, by one. Next, if the counter is greater than an over−threshold indicator, OT, then the proposed algorithm recalculates an optimal data burst size through some equations which will be shown in this paper, and reset OI.

As traffic load is changed, core nodes send some feedback messages to edge nodes to reduce the size of data bursts using Algorithm 1, and edge nodes adjust the size of data bursts to obtain the optimal size of data bursts through Algorithm 2.

On the other hand, in addition to the above two algorithms, Algorithms 1 and 2, another algorithm is needed to increase the size of data bursts. If data bursts are successfully transmitted, then stepping up the size of data bursts is required to increase the efficiency of OBS networks. Algorithm 3 shows the routine for core nodes to increase it. As shown in Algorithm 3, if BC_c^k is less than BC_t^k, then an under−counter indicator for class k, UC_k, is increased by one. Also, if UC_k is greater than the under−threshold−counter indicator for class k, UT_k and FC_k is greater than the minimum number of FDLs of class k, $FC_k^{\min}$, then FC_k is decreased by one and FC_n is increased by one. However, if FC_k is less than or equal to $FC_k^{\min}$, then a core node transmits some increasing request messages to edge nodes.

3.2 Performance Analysis

In order to analyze the performance of our proposed scheme, some assumptions are used as follows: the arrival process for IP packets is Poisson with rate λ_p [packets/second] and the average packet length is exponentially distributed with mean $1/\mu_p$, where μ_p [packets/second] is service rate for IP packets, and each core node has m FDLs. Also, It is assumed that there are n traffic classes, in which class 1 is the highest class and class n is the lowest one, best effort traffic class, and all FDLs of each core node are divided into n sub−FDL groups. Sub−FDL groups $1,2,\ldots,n$ are preassigned to traffic classes $1,2,\ldots,n$, respectively. The number of FDLs of each sub−FDL group is changed through our proposed algorithms as traffic conditions are changed. When there are two traffic classes i and j $(i<j)$, the higher class i can use sub−FDL groups from i to n including sub−FDL group j, but the lower class j can only reserve sub−FDL groups from j to n and cannot use any higher sub−FDL groups from 1 to $j-1$. In our analysis, any wavelength converters

are not considered.

First, the blocking probability for each class at core nodes is analyzed. In order to maintain the FIFO ordering discipline, when $i < j$, the shortest FDL length of sub-FDL group j should be longer than the longest one of sub-FDL group i. Now, the current loss probability of class i, denoted BC_c^i $(i = 1, 2, \ldots, n)$, is derived. In order to obtain BC_c^i, the blocking probability of each group, BG_i $(i = 1, 2, \ldots, n)$ affected by the state of each delay line in sub-FDL group i and data burst arrival rate for the group [13], is considered. BG_i can be displayed as follows:

$$BG_i = f(FG_i, \lambda_i^t). \tag{1}$$

where FG_i for sub-FDL group i composed of t delay lines is the set that consists of $\{\mathrm{f}_\mathrm{i}^1, \mathrm{f}_\mathrm{i}^2, \ldots, \mathrm{f}_\mathrm{i}^\mathrm{n}\}$, where state $f_i^k (1 \le k \le t)$ is reached by each FDL buffer when a data burst is switched to delay line k in the sub-FDL group, and λ_i^t is total data burst arrival rate for the group, which is given as follows:

$$\lambda_i^t = \lambda_i + \sum_{k=1}^{i-1} \lambda_k \prod_{j=k}^{i-1} BG_j, \tag{2}$$

where the first part is the data burst arrival rate of traffic class i for sub-FDL group i and the second part is the arrival rate from upper traffic classes from 1 to $i-1$, which are not transmitted through upper sub-FDL groups.

Now, let us find the blocking probability for each sub-FDL group. Data bursts of traffic class $n-1$ are blocked when they can not find any available delay lines in sub-FDL groups $n-1$ and n because blocked data bursts of the class can reserve lower sub-FDL group n. Also, Traffic class $n-2$ are discarded when it can not reserve any delay lines in sub-FDL groups from $n-2$ to n. Likewise, class 1 can not be transmitted only when all sub-FDL groups do not have any available delay lines. So, BC_c^i can be given as follows:

$$BC_c^i = \prod_{j=i}^{n} BG_j. \tag{3}$$

BC_c^i is calculated at each core node. If the value is over the target blocking probability of class i ,BC_t^i, then our algorithm will first control the number of FDLs in the sub-FDL group. However, if BC_c^i is still bigger than BC_t^i, then it sends some adjustment requests to edge nodes transmitting those data bursts to control the size of data bursts.

Now, let us adjust the size of data bursts. In order to calculate an optimal size of data bursts, blocking probabilities need to be calculated at edge nodes using some traffic information from core nodes. As shown in [22], blocking probabilities are affected by the average data burst arrival rate λ_b [data bursts/second], the average data burst service rate μ_b [data bursts/second], the number of FDLs m, and the basic delay unit D. So, blocking probabilities at edge nodes BP_e can be expressed as follows:

$$BP_e = f(\lambda_b, \mu_b, m, D). \tag{4}$$

In Eq. 4, μ_b [data bursts/second] can be easily obtained. In order to transmit data bursts, the total size of aggregated IP packets should be bigger than the threshold value for data bursts. So, μ_b is given as follows:

$$\mu_b = \frac{\mu_p W_c}{S_b \mu_p + W_c}, \tag{5}$$

where S_b [bits] is the threshold value for data bursts, W_c [bits/second] is the bandwidth of each channel, and $1/\mu_p$ [second] is the mean length of IP packets with exponential distribution. Also, in order to be in steady-state, the rate of an input channel is the same as that of an output channel as follows]: $\lambda_p/\mu_p = \lambda_b/\mu_b$. From the proportional expression and Eq. 5, λ_b is given as follows:

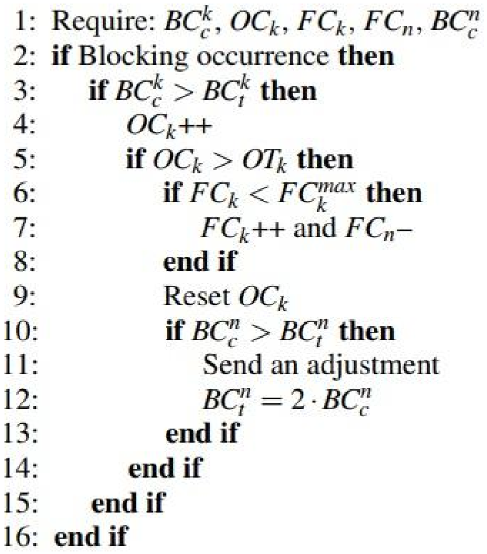
$$\lambda_b = \frac{\lambda_p W_c}{S_b \mu_p + W_c}. \qquad (6)$$

When given traffic conditions (λ_b and μ_b), the number of FDLs (m), and a fixed basic delay unit D, optimal blocking probabilities can be obtained using Eq. 4. However, core nodes have different basic delay units and edge nodes can not know the basic delay unit values of core nodes. So, when the traffic conditions change, a new set of FDLs with a different optimal basic delay unit should be used, but the FDLs are fixed hardware components. Instead, in order to find optimal blocking probabilities, the threshold value for data bursts, S_b, should be recalculated because some traffic conditions such as λ_b and μ_b can be affected by the new calculated S_b.

From Eq. 5, the average data burst size is given as follows:

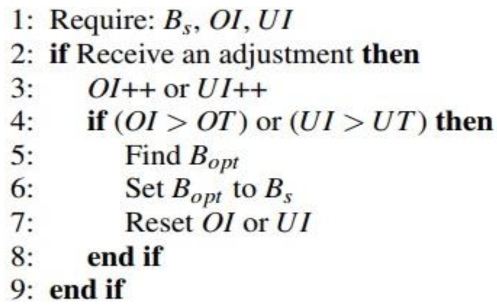
$$E[L_b] = \frac{1}{\mu_b} = \frac{S_b \mu_p + W_c}{\mu_p W_c} = \frac{S_b}{W_c} + \frac{1}{\mu_p} = L, \quad (7)$$

where $E[X]$ is the expected value of a random variable X and L_b [second] is a random variable representing data burst lengths. Because the data assembly process is threshold-based, $E[L_b]$ is composed of two parts: one is a fixed portion (the first part in Eq. 7) and the other is a variable portion (the second part in Eq. 7).



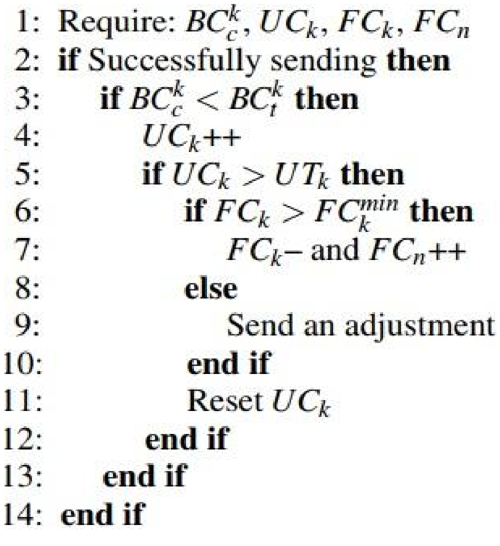
```
Require: $BC_c^k$, $OC_k$, $FC_k$, $FC_n$, $BC_c^n$
if Blocking occurrence then
    if $BC_c^k > BC_t^k$ then
        $OC_k$++
        if $OC_k > OT_k$ then
            if $FC_k < FC_k^{max}$ then
                $FC_k$++ and $FC_n$−
            end if
            Reset $OC_k$
            if $BC_c^n > BC_t^n$ then
                Send an adjustment
                $BC_t^n = 2 \cdot BC_c^n$
            end if
        end if
    end if
end if
```

Fig. 2. Burst size decreasing request algorithm.



```
Require: $B_s$, $OI$, $UI$
if Receive an adjustment then
    $OI$++ or $UI$++
    if ($OI > OT$) or ($UI > UT$) then
        Find $B_{opt}$
        Set $B_{opt}$ to $B_s$
        Reset $OI$ or $UI$
    end if
end if
```

Fig. 3. Burst size decision algorithm.



```
Require: $BC_c^k$, $UC_k$, $FC_k$, $FC_n$
if Successfully sending then
    if $BC_c^k < BC_t^k$ then
        $UC_k$++
        if $UC_k > UT_k$ then
            if $FC_k > FC_k^{min}$ then
                $FC_k$− and $FC_n$++
            else
                Send an adjustment
            end if
            Reset $UC_k$
        end if
    end if
end if
```

Fig. 4. Burst size increasing request algorithm

4. Performance Evaluation

In this section, extensive simulations and numerical analyses are carried out using Riverbed Modeler (former OPNET) network simulation tool. Some parameters are assumed as follows: the bandwidth of each channel, W_c, is 1 Gbps, the initial threshold value is set to 200 Kbits, the arrival process for IP packets is Poisson with rate λ_p, the average IP packet length is exponentially distributed with $1/\mu_p$, and a threshold based assembly process is considered.

In order to show the validity of our proposed algorithm dynamically controlling data burst length, it is assumed that there three traffic classes, where class 1 is the highest class and class 3 is the lowest one. Also, it is assumed that FDLs are divided into three sub-FDL groups because three traffic classes are considered, and that sub-FDL group 1 is preassigned to traffic class 1, sub-FDL group 2 to traffic class 2, and sub-FDL group 3 to traffic class 3. The FDLs in sub-group 1 are used only by the data bursts of class 1, sub-group 2 by class 1 and 2, sub-group 3 by all classes. Moreover, it is assumed that the distribution ratio of class 1, class 2, class 3 is 10%, 20%, 70%, respectively, and that the target loss probability for class 1 is 10^{-4} and for class 2 is 10^{-3}.

Figure 5 shows the variation of blocking probabilities when the basic delay unit (D), which is the relative length to date burst size and is decided through the proposed algorithms, changes. Also this figure shows that as the offered load increases, the relative length should be changed too.

Now, let us compare our new algorithm with our old one. Two algorithms are compared in Fig. 6. As shown in Fig. 6(a), when the offered load is high, it is known that class 3 traffic (best effort service) is hardly transmitted. In addition to that, when the offered load is 0.9, the loss probability of class 2 traffic is higher than the target loss probability for that class.

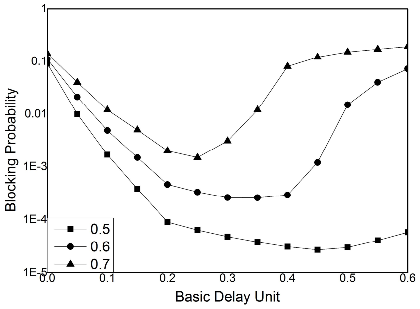
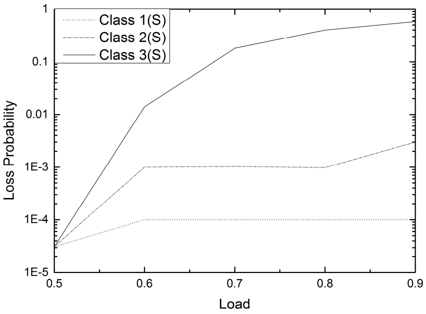


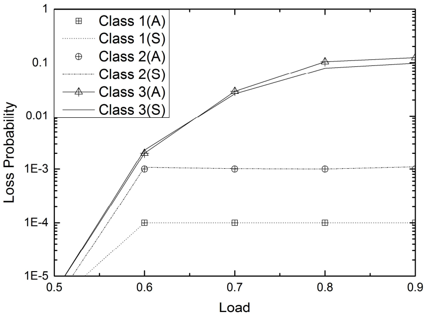
Fig. 5. Variation of blocking probabilities when changing the basic delay unit value D.

However, as shown in Fig. 6(b) (Class 1 (A) means the loss probabilities for class 1 by analyses and Class 1 (S) by simulations), it is known that the loss probabilities of class 3 traffic are quite lower then those of our old algorithm. Moreover, while in Fig. 6(a), when the offered load is 0.9, the old algorithm can not match with the target loss probability, our new algorithm can exactly step on the target loss probability for class 2.

In fact, the biggest merit of our new algorithm is shown in Fig. 7. The figures show the variation of the number of FDLs for sub-FDL groups. In the figures, class 1 (F) means that the results were calculated by our old algorithm using fixed threshold value, and class 1 (V) uses our new algorithm dynamically controlling data burst lengths. From Fig. 7(a), in old algorithm case, the variation of the number of FDLs for sub-FDL group 1 is quite small because the sub-group is only used by traffic class 1, but the change of sub-FDL group 2 is pretty big because the sub-group is used by traffic class 2 and traffic class 1 not to be transmitted through sub-FDL group 1. On the other hand, from the figure, it is shown that when our new algorithm is used, the amount of variation is quite smaller than that of old algorithm case.

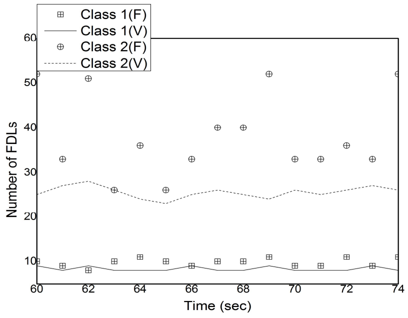


(a) Old algorithm case.

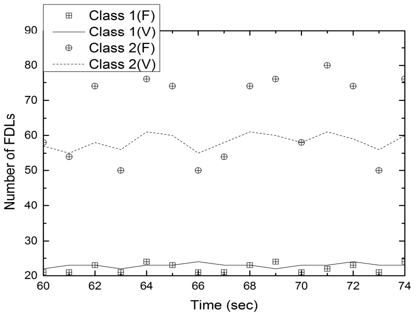


(b) New algorithm case

Fig. 6. Loss probabilities for each class when 64 FDLs are used.



(a) Case of 64 FDLs



(b) Case of 128 FDLs

Fig. 7. Number of FDLs for each sub-FDL group when the offered load is 0.7.

5. Conclusion

In order to overcome some problems of our old algorithm, a new service differentiation algorithm was proposed. Also, in order to achieve service differentiation in OBS networks, our new algorithm can dynamically control the number of FDLs in sub-groups and adjust the threshold value according to traffic conditions. Moreover, when our new algorithm was adopted, it was known that the fluctuation of the number of FDLs in each sub-group was considerably reduced.

Because of the degree of the complexity, the feed-forward type buffering architecture was taken into account. However, the development of the optical buffering technologies leads to the grows of shared type architecture. Therefore, the utilization of the shared architecture leaves as further works.

REFERENCES

[1] Q. Zhang & C. Neal. (2011). TCP over Optical Burst-switched Networks with Controlled Burst Retransmission. *Photonic Network Communications, 22(3),* 299-312.
DOI : 10.1007/s11107-011-0329-8

[2] V. Vinod. (2002). Burst Segmentation: An Approach for Reducing Packet Loss in Optical Burst-switched Networks. *Proceedings of ICC 2002, (5),* 2673-2677.
DOI : 10.1109/ICC.2002.997328

[3] R. Poorzare & S. Abedidarabad. (2017). Optimizing Optical Networks by Using CWN Algorithm. *Review of Computer Engineering Studies, 4(3),* 98-102.
DOI : 10.18280/rces.040303

[4] F. Testa & L. Pavesi. (2018). *Optical Switching in Next Generation Data Centers*. New York : Springer.

[5] Z. Zhou, T. Lin, K. Thulasiraman, G. Xue & S. Sahni. (2015). Cross-layer Network Survivability under Multiple Cross-layer Metrics. *Journal of Optical Communications and Networking, 7(6),* 540-553.
DOI : 10.1364/JOCN.7.000540

[6] Z. Zhou, T. Lin, K. Thulasiraman & G. Xue. (2017). Novel Survivable Logical Topology Routing by Logical Protecting Spanning Trees in IP-over-WDM Networks. *IEEE/ACM Transactions on Networking, 25(3),*

1673-1685.
DOI : 10.1109/TNET.2016.2639362

[7] L. Xu, Q. Guo, T. Yang & H. Sun. (2019). Robust Routing Optimization for Smart Grids Considering Cyber-Physical Interdependence. *IEEE Transactions on Smart Grid, 10(5),* 5620-5629.
DOI : 10.1109/TSG.2018.2888629

[8] A. D. Hossain & M. Hossain. (2018). Burst Mode Optical Receiver with 10 ns Lock Time Based on Concurrent DC Offset and Timing Recovery Technique. *Journal of Optical Communications and Networking, 10(2),* 65-78.
DOI : 10.1364/JOCN.10.000065

[9] C. Qang, H. Cao, S. Yang, J. Guo, H. Gui & J. Wu. (2018). Decision Tree Classification Based Mix-flows Scheduling in Optical Switched DCNs. *Proceedings of OFC 2018,* 1-3.
DOI : 10.1364/OFC.2018.W1C.4

[10] S. Kim, N. Kim & M. Kang. (2002). Contention Resolution for Optical Burst Switching Networks Using Alternative Routing. *Proceedings of ICC 2002, (5),* 2678-2681.
DOI : 10.1109/ICC.2002.997329

[11] R. Alex & C. I. Oliver. (2007). A Survey of IP and Multiprotocol Label Switching Fast Reroute Schemes. *Computer Networks, 51(8),* 1882-1907.
DOI : 10.1016/j.comnet.2006.09.010

[12] D. Avranil & B. Paramita. (2008). Performance Evolution in Optical Burst Switched Networks. *Proceeding of WOCN 2008,* 1-5.
DOI : 10.1109/WOCN.2008.4542478

[13] M. Yoo & C. Qiao. (2000). The Effect of Limited Fiber Delay Lines on QoS Performance of Optical Burst Switched WDM Networks. *Proceedings of ICC 2000, (2),* 974-979.
DOI : 10.1109/ICC.2000.853643

[14] S. Ramamurthy & B. Mukherjee. (1999). Survavable WDM Mesh Networks. I. Protection. *Proceedings of INFOCOM 1999, 2,* 744-751.
DOI : 10.1109/INFCOM.1999.751461

[15] S. Vegesna. (2001). *IP Quality of Service,* Hoboken : Cisco Press

[16] S. S. Chawathe. (2018). Analysis of Burst Header Packets in Optical Burst Switching Networks. *Proceedings of NCA 2018,* 1-5.
DOI : 10.1109/NCA.2018.8548071

[17] R. D. Doverspike, G. Sahin, J. L. Strand & R. W. Tkach. (1999). Fast Restoration in a Mesh Network of Optical Cross-connects. *Proceedings of OFC 1999, 1,* 170-172.
DOI : 10.1109/OFC.1999.767828

이 용 규(Yonggyu Lee) [정회원]

- 1996년 2월 : 전북대학교 컴퓨터공학과(공학사)
- 2000년 2월 : 한국과학기술원 정보통신공학과(공학석사)
- 2008년 2월 : 한국과학기술원 정보통신공학과(공학박사)
- 2015년 8월 ~ 현재 : 건양대학교 인문융합학부 교수

- 관심분야 : Optical Internet, QoS Guaranteed Netwok
- E-Mail : lyonggyu@konyang.ac.kr