Journal of Digital Convergence
Vol. 20. No. 4, pp. 459-468, 2022

ISSN 2713-6434 / eISSN 2713-6442
https://doi.org/10.14400/JDC.2022.20.4.459

Analysis of Operation Areas for Automatically Tuning Burst Size-based Loss Differentiation Scheme Suitable for Transferring High Resolution Medical Data

Yonggyu Lee
independent researcher

고해상도 의학 데이터 전송에 적합한 자동 제어 버스트 크기 기반 손실 차등화 기법을 위한 동작 영역 분석

이용규
독립연구자

Abstract In medical area, very high resolution images, which is loss sensitive data, are used. Therefore, the use of optical internet with high bandwidth and the transmission of high realiability is required. However, according to the nature of the Internet, various data use the same bandwidth and a new scheme is needed to differentiate effectively these data. In order to achieve the differentiation, optical delay line buffers are used. However, these buffers is constructed based on some optimal values such as the average offered load, measured data burst length, and basic delay unit. Once the buffers are installed, they are impossible to reinstall new buffers. So, the scheme changing burst length dynamically was considered. However, this method is highly unstable. Therefore, in this article, in order to guarantee the stable operation of the scheme, the analysis of operation conditions is performed. With the analysis together with the scheme, high resolution medical data with the higher class can transmit stably without loss.

Key Words : Optical Internet, Fiber Delay Line, Operation Conditions, QoS, OBS

요 약 의료 현장에서는 매우 고해상도의 이미지를 사용하고 있으며, 이는 손실에 매우 민감한 정보이다. 이에 따라 높은 대역폭뿐만 아니라 고신뢰성 전송을 제공할 수 있는 광 인터넷의 활용이 요구되고 있다. 그러나 인터넷의 특성상 다양한 종류의 데이터가 동일한 대역폭을 활용하게 되고, 이를 효과적으로 차별화할 수 있는 수단이 요구되고 있다. 이를 위해 광 지연 라인 버퍼가 많이 활용되고 있다. 그러나, 이러한 버퍼는 제공 부하, 측정된 데이터 버스트 크기, 기본 지연 유닛 등과 같은 최적값을 이용해 구성된다. 광 버퍼는 한 번 설정되면 변경할 수 없다. 그러므로 데이터 버스트 크기를 동적으로 변경시키는 방법이 활용되고 있다. 그러나 동적으로 버스트의 길이를 변화시키는 것은 상당한 불안정성을 내포하고 있다. 이에 본 논문에서는 안정적인 동작을 보장할 수 있는 동작 조건을 분석하고자 한다. 본 논문의 기법을 활용해 높은 우선순위의 고해상도 의료 데이터를 손실 없이 안정적으로 전송할 수 있다.

주제어 : 광 인터넷, 광 지연 라인, 동작 조건, 서비스 품질, OBS

*Corresponding Author : Yonggyu Lee(lyonggyu@gmail.com)
Received February 18, 2022 Revised March 13, 2022
Accepted April 20, 2022 Published April 28, 2022

1. Introduction

In order to satisfy the requirement for a huge amount of transmission bandwidth for various multimedia applications, various up-to-date optical switching technologies such as optical burst switching (OBS), optical packet switching (OPS), and optical label switching (OLS) have been studying.

Among those technologies, OBS with strong separation between signaling and data transmission is very promising technology. In OBS, signaling transmission is transmitted through the electrical channels using optical-electrical conversions, while data transfer is sent through the all optical channels without any conversions. In fact, optical Internet still has many defects such optical protocol, optical memory, and optical buffer and so on. Because of these problems, in optical Internet core nodes have severe bottlenecks at intermediate nodes. However, because of the separation between signaling and data transmission in OBS networks, the intermediate switching nodes can remove or relax electrical bottlenecks and easily guarantee the required QoS without any buffering [1-5].

Because of the advantages of OBS technology, many differentiation schemes still have been studying. In [6], in order to achieve loss and delay differentiation, a proportional QoS scheme was proposed. In [7], a differentiation method based on offset times was proposed. In [8], a preemptive differentiation scheme using burst segmentation technology was studied. In [9], in order to maintain the preassigned differentiation, a usage profile based method was displayed. In [10], forwarding resource reservation method using a linear predictive filter was proposed. In [11], if data bursts are arrived at intermediate switches without any proper signaling information and they violate the predefined loss probabilities, then they do not have any proper buffering and are dropped in a probabilistic manner. In [12, 13], GMPLS-based differentiation schemes were illustrated. In those articles, QoS-required services are sent through label switches path, but the rest services without requiring QoS are transmitted on the rest network capacity. In [14], an analytically differentiated model was proposed. In [15], a scheduling mode suitable for delay critical applications was proposed. In [16], a fiber delay line (FDL) based QoS method was proposed. In [17], a shared type FDL based QoS scheme was proposed. In [18], several data burst scheduling algorithms bases on FDL was proposed.

In general, when OBS core and edge nodes are designed, the FDLs are installed with some fixed parameters such as the average data burst length, basic delay unit, the number of FDLs and so on at core nodes, and the arrival packet rate, the average packet length, available memory and on on at edge nodes. If one of those parameters is changed at core nodes, the fixed FDLs should be altered, but it is fixed hardware component. Therefore, in order to solve the problem to require new FDLs, a new dynamic burst length controlling algorithm-based differentiation scheme based on the feed-forward type FDL architecture was proposed in [19].

However, in [19] the author could not obtain some stable working conditions. If there are a number of delay lines, data burst loss can be zero. Also, the large delay lines will increase the cost and the size of switches. So, in order to achieve the trade-off between cost and size of switches and stable operation, some feasibility conditions are calculated in this article.

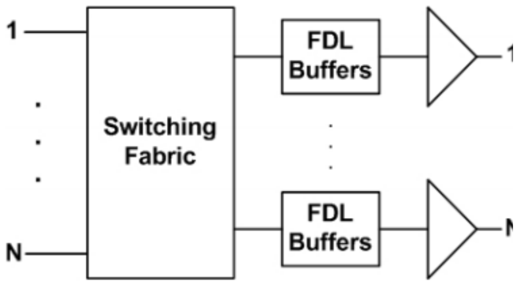
This article is organized as follows. Section 2 illustrates buffering architectures. Section 3 shows the summarized algorithm, which is proposed previously, and some feasibility conditions. Section 4 displays the performance evaluation for those feasibility conditions. Finally, Section 5 closes this article with conclusion.

2. Buffering Architecture

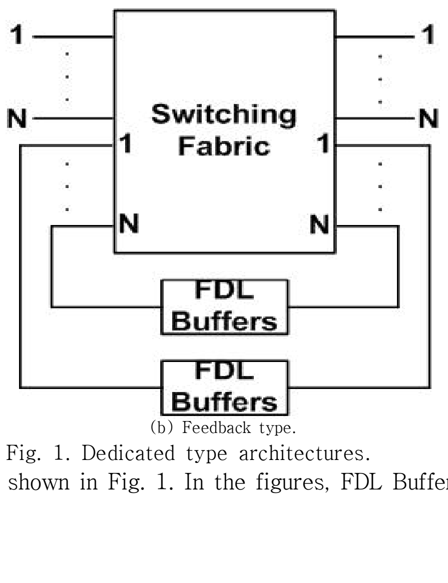
Because there is no optical buffer memory, in order to accomplish the effect of memory, optical fiber is used in OBS nodes. In order to compose optical memory, called fiber delay line (FDL), through optical fiber, a certain basic delay unit value, D, is considered. A FDL is simply consecutive multiples of the D value. Because the FDLs can not store light signal signals, they can introduce some small delay to signals. Because of this feature, in order to imitate the electrical memory systems, a few buffering architectures based on FDLs have been proposed as follows:

- according to burst flow direction: feedback and feed−forward,
- according to resource sharing: shared and dedicated.

Two dedicated type FDL buffering configurations



(a) Feed−forward type.

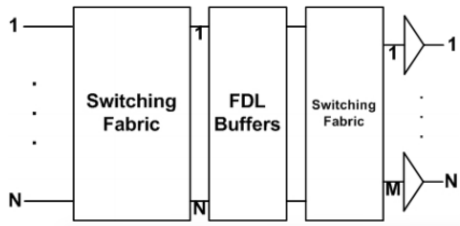


(b) Feedback type.

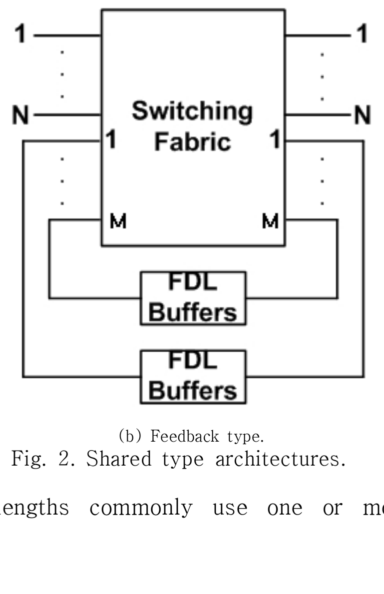
Fig. 1. Dedicated type architectures.

are shown in Fig. 1. In the figures, FDL Buffers, or called FDL bank, are consisted of many FDLs, each of which is consecutive multiples of a certain D value. In Fig. 1. (a) is dedicated type feed−forward configuration and (b) is dedicated type feedback configuration. From the figures, two configurations need the same number of FDL buffers, but feed−forward configuration need quite smaller switching fabric than feedback configuration as shown in Fig. 1. However, the architecture of switching fabric of feedback configuration is much simpler than that of feed−forward configuration.

Other FDL architectures are shown in Fig. 2, which are called shared type FDL configurations. As shown in Fig. 2, the shared types have much fewer FDL buffers than the dedicated types. In fact, in the dedicated configurations, each wavelength has its own FDL buffers block, but in the shared configurations, several or all



(a) Feed−forward type.



(b) Feedback type.

Fig. 2. Shared type architectures.

wavelengths commonly use one or more FDL buffers blocks. Also, the shared types have two configurations: feed-forward shown in Fig. 2(a) and feedback configurations shown in Fig. 2(b).

In OBS networks, besides FDL architectures, the performance of data burst forwarding can be affected by other factors such as the distribution of the length of each FDL in FDL buffers and data burst forwarding policy.

FDL buffers can be constructed as degenerate or non-degenerate method. In degenerate method, when there are two FDLs i and j, which the length of jth FDL is longer than that of ith FDL, their lengths are configured as follows:

- ith FDL: $i \times D$,
- jth FDL: $j \times D$,

which D is the basic delay unit value. On the other hand, in non-degenerate method, the length of each FDL is randomly increased, which means that non-degenerate method does not require any D value.

The other import factor is the data burst forwarding policy, data burst scheduling policy. In general, OBS nodes use two types of scheduling policies: one is void-filling and the other is non-void-filling. Void means an interval between two consecutive data bursts. In OBS networks, voids occurs because FDLs can offer only a fixed delay value. In void-filling method, incoming data bursts can be transferred to any available FDLs as long as some contentions do not happen at those FDLs. So, every OBS node can minimize the occurrence of voids. However, this policy can change the sequences of data bursts. On the other hand, in non-void-filling policy, even though there are very big voids between two data bursts, any incoming data bursts can not use those voids. Therefore while void-filling method is much better performance than non-void-filling policy, the void-filling scheme may need complex sequencing overhead.

In this article, in order to evaluate some feasibility conditions, dedicated type feed-forward architecture, Fig. 1(a), is selected.

3. Feasibility Analysis

In this section, after summarizing briefly the proposed algorithm [19], the analysis of the feasibility conditions is performed.

3.1 Summarized Algorithm

In this subsection, the previously proposed algorithm is summarized. The idea of the algorithm is very simple. If a sub-FDL group has enough FDLs to transmit data bursts, the algorithm decreases the number of FDLs for the group, but if the group does not enough FDLs, the algorithm increases the number of FDLs. In other words, when data burst contentions occur at class k, each core node computes the blocking probability for class k, and the blocking probability is compared with the given target blocking probability for class k. If the calculated blocking probability is greater than the target blocking probability, then the algorithm increases the over-counter of class k by one. Also, if the over-counter of class k is greater than the over-threshold-counter of the class, then the number of FDLs of group k is compared with the upper limit of the number of FDLs for group k. If the number of FDLs of group k is less then the upper limit of the number of FDLs of the group, then the algorithm increases the number of FDLs of the group by one and decreases the number of FDLs for best-effort traffic class by one. Then, the over-counter of class k is initialized to zero, and if the blocking probability of class n, best-effort service, is greater than the blocking probability of the class, then core nodes transmits some feedback informations which include the information of the contenting data bursts to the edge nodes as shown in Algorithm 1 in Fig. 3.

```
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 · BC_c^n
            end if
        end if
    end if
end if
```

Fig. 3. Burst size decreasing request algorithm.

As soon as edge nodes receive the adjustment messages generated by Algorithm 1, they adjust the length of data bursts using Algorithm 2 in Fig. 4, data burst size controlling algorithm. As soon as edge nodes receive the information, they increase their own over-indicator by one. Then, an optimal data burst size is recalculated by the algorithm if the indicator is greater than over-threshold, and the algorithm initializes the over-indicator to zero. Core nodes transmit some network status information to edge nodes to adjust the length of data bursts through Algorithm 1 when traffic conditions are changed. Then, edge nodes perform Algorithm 2 to get the optimal length of data bursts.

```
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. 4. Burst size decision algorithm.

Besides the above two algorithms, data burst size increasing algorithm is needed because if network has enough resources, then in order to achieve the high throughput, the data burst length should be increased. If data bursts are well transferred through pre-allocated channels, then edge nodes should increase the length of data bursts to enhance the efficiency of OBS networks. However edge nodes do not know the status of OBS networks. So, core nodes should send network status information to edge nodes as shown Algorithm 3 in Fig. 5.

```
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. 5. Burst size increasing request algorithm

If the blocking probability for class k is less than the given target blocking probability for the class, then an under-counter of the class is increased by one as Algorithm 3. Next, if the under-counter of class k is greater than the under-threshold-counter of the class, and the number of FDLs of group k is greater than the minimum number of FDLs for the class, then the number of FDLs of the group is decreased by one and the number of FDLs of best-effort class is increased by one. Then, if the number of FDLs of group k is less than or equal to the minimum number of FDLs of class k, then edge nodes receive increasing request message from core nodes.

3.2 Feasibility Conditions

In this subsection, the feasibility conditions for the algorithm are analyzed.

In the algorithm, n traffic classes is considered, in which the highest class is 1, and the lowest class is n. The number of sub-FDL groups is the same number of traffic classes, n, and each sub-group is preassigned to each class, respectively. When there are two traffic classes i and j, in which i is the higher class than j, class i data bursts take sub-groups from i to n, but class j data bursts only utilize sub-groups from j to n. The lower class j data bursts can not occupy sub-groups for the higher class i.

The blocking probability for class i (BP_c^i) is gotten as follows:

$$BP_c^i = \prod_{j=i}^{n} BG_i, \quad (1)$$

where BG_i is the blocking probability for sub-group i, and BG_i can be calculated as follows:

$$BG_i = f\left(SG_i, \lambda_i + \sum_{k=1}^{i-1} \lambda_k \prod_{j=k}^{i-1} BG_j\right), \quad (2)$$

where SG_i is sub-FDL group i and λ_i is the data burst arrival rate of class i.

Also, the average length for data bursts is given by the algorithm as follows:

$$L = S_b B_c^{-1} + \mu_p^{-1}, \quad (3)$$

where S_b is the data burst length threshold value and B_c is the bandwidth of each channel.

From Eq. 1, the blocking probability for class 1 is given as follows:

$$BP_c^1 = \prod_{i=1}^{n} BG_i. \quad (4)$$

In order to fulfill the given target blocking probability of the class, the previous equation can be rewritten as follows:

$$BP_c^1 = BG_1 \prod_{i=2}^{n} BG_i, \quad (5)$$

and the blocking probability for class 2 is given like that:

$$BP_c^2 = \prod_{i=2}^{n} BG_i = BG_2 \prod_{i=3}^{n} BG_i. \quad (6)$$

From Eqs. from 4 to 6, the blocking probability for class k can be given as follows:

$$BP_c^k = BG_k \prod_{i=k+1}^{n} BG_i. \quad (7)$$

From Eq. 7, the blocking probability for sub-group j is given as follows:

$$BG_j = \frac{BP_c^j}{BP_c^{j+1}}. \quad (8)$$

The number of delay lines, FC_j, for sub-FDL group j can be found by the proposed algorithm and using the obtained FC_j, the highest blocking probability can be calculated because the fewer FC_j, the higher blocking probability, while the more FC_j, the lower blocking probability. From the fact, the minimum number of delay lines for sub-FDL groups which are satisfied with the target blocking rates are given from Eq. 7. So, when given FC_j, at least the following inequality should be held:

$$BG_j \le \frac{BP_c^j}{BP_c^{j+1}}. \quad (9)$$

Now that the minimum number of delay lines of sub−FDL group j are calculated, let us find the maximum value. In order to transmit data bursts belonging to class k through sub−FDL group k, the sub−group should have at least one available delay line in the sub−group. This can be applied to all sub−groups. From this fact, the maximum number of delay lines of sub−FDL group j are given as follows:

$$FC_j = \sum_{k=1(k \neq j)}^{n} FC_k^{\min}. \quad (10)$$

4. Performance Evaluation

In this section, simulations and numerical analysis are performed using Riverbed Modeler network simulation tool. In order to evaluate those simulations and analysis, Some assumptions are made as follows:

- each channel bandwidth: 1 Gpbs,
- initial threshold length: 200 Kbits,
- IP packet process: Poisson process,
- distribution of the average IP packet length: exponential,
- burst aggregation method: threshold based process,
- target blocking probability for class 1 and 2: 10^{-4} and 10^{-3}, respectively, and
- burst traffic distribution for class 1, 2, and 3: 10%, 20%, and 70%, respectively.

In order to prove the validation of the proposed algorithm, this article uses three traffic classes. Among these classes, class 1 and 3 are the highest and the lowest classes, respectively. Also, 3 sub−FDL groups are utilized because 3 classes are considered. Before simulations and analysis are started, each sub−group is preassigned as follows: sub−group 1 to data burst traffic class 1, sub−group 2 to traffic class 2, and sub−group 3 to traffic class 3, respectively.

First, in order to find the optimal D value for simulations and analysis, when various offered load conditions are given, according as the D value changes, the variation degrees are shown in Fig. 6. From the figure, it is known that if some traffic conditions such as the offered load and the basic delay unit value, then the blocking probabilities are changed, too. This shows that according as some traffic conditions change, the FDL bank should be followed.

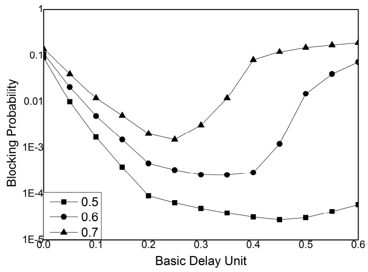


Fig. 6. Variation of blocking probabilities when changing the basic delay unit value D.

Next, in order to find the advantages of the new algorithm, from simulations, Table 1 (S.D. means standard deviation.) shows the statistical values for two algorithms. As shown in the table, it is known that the S.D. values are quite big when old algorithm was used. On the other hand, when the new algorithm was used, the table shows that the S.D. values are quite small. Moreover, while the number of FDLs of sub−group 1 was the almost same, it was known that the performance of sub−group 2 was quite improved. This means that the stability of the newly proposed algorithm was improved a lot.

Table 1. Amount of the change of the number of FDLs for each sub-FDL 2group when 64 FDLs are used.

Old Algorithm				New Algorithm			
Mean		S.D.		Mean		S.D.	
1	2	1	2	1	2	1	2
9.8	38.4	0.941	9.195	8.333	25.5333	0.488	1.302

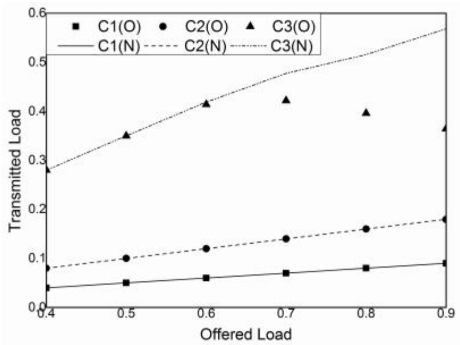


Fig. 7. The Variation of transmitted load according to the offered load.

Fig. 7 shows another merit of the new algorithm. In the figure, 64 FDLs are used. As shown in the figure, in the case of the higher classes, both old algorithm (represented as C1 (O) and C2 (O)) and new one (represented as C1 (N) and C2 (N)) transmit their data bursts well. However, in the lowest class, the old algorithm loses quite a lot of its data bursts, while the new one does transfer its traffic well. As shown in Table 1, the higher classes take the most resources to transfer their data bursts. The table shows that the old algorithm uses 49 FDLs, while the new one utilizes only 35 FDLs. Because of that, the new algorithm can give more FDLs to the lowest class and this fact illustrates why the new algorithm can transmits more data bursts than the old one.

Also, the variations of the threshold values are illustrated. From Fig. 6, it is clearly known that according to the offered load, the basic delay unit D should be changed. Especially, the figure shows that according as the offered load increases, the D values decrease. From the previous result and Eq. 3, it is known that according as the offered load increases, the threshold value is reduced, too. Namely, From Fig. 6 and Eq. 3, it is known that according to the increment of the offered load, the basic delay unit D and the threshold value S_b should be diminished as shown in Fig. 8.

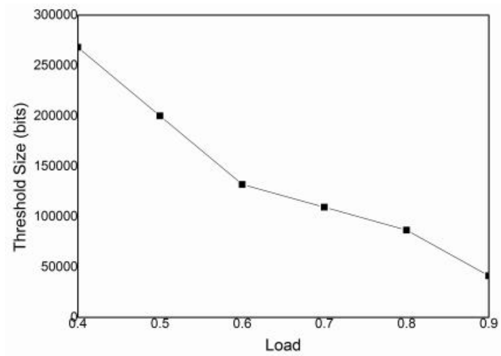


Fig. 8. Variation of the threshold values when the offered load changes.

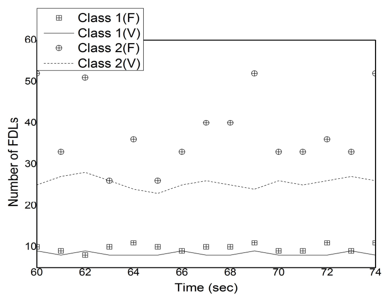


Fig. 9. The comparison of old algorithm (F) and new algorithm (V), when 64 FDLs are considered.

Lastly, Fig. 9 shows the difference between old algorithm and new one. The old algorithm has quite broad variation, while the new one quite small fluctuation. From the figure, it is known that the new algorithm enhance the stability of network.

5. Conclusion

In order to accomplish service differentiation for OBS networks, a new scheme was proposed using feed-forward type output buffering architecture in the previous work. Through the algorithm, the length of data bursts are automatically tuned based on the status of network. However, the work did not evaluate steady state conditions for the algorithm. Therefore, some operation conditions are evaluated through extensive simulations and numerical analysis. This paper shows that if the conditions are considered, the algorithm improves considerably the stability.

REFERENCES

[1] Y. Yinghua, S. Dixit & M. Ali. (2000). On Joint Protection/Restoration in IP-centric DWDM Based Optical Transport Networks. *IEEE Communications Magazine, 38(6),* 174-183.
DOI : 10.1109/35.846091

[2] 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

[3] 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

[4] 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

[5] 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

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

[7] 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

[8] Z. Zhou, T. Lin & K. Thulasiraman. (2017). Survivable Cloud Network Design against Multiple Failures through Protecting Spanning Trees. *Journal of Lightwave Technology, 35(2),* 288-298.
DOI : 10.1109/JLT.2016.2637352

[9] 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

[10] R. Cohen & G. Nakibly. (2017). Restorable Logical Topology in the Face of No or Partial Traffic Demand Knowledge. *IEEE/ACM Transactions on Networking, 24(4),* 2074-2085.
DOI : 10.1109/INFOCOM.2014.6848099

[11] 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

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

[13] M. Xu, K. Naik & K. Thulasiraman. (2020). Fault Tolerance of Hypercube like Networks: Spanning Laceability under Edge Faults. *Theoretical Computer Science, 835(2),* 44-57.
DOI : 10.1016/j.tcs.2020.05.049

[14] 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

[15] 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

[16] M. Goyal et al. (2012). Improving Convergence Speed and Scalability in OSPF: A Survey. *IEEE Communications Surveys & Tutorials, 14(2),* 443-463.
DOI : 10.1109/SURV.2011.011411.00065

[17] P. Li & M. Xu. (2017). The Super Spanning Connectivity of Arrangement Graphs. *International Journal of Foundations of Computer Science, 28(8),* 1047-1072.
DOI : 10.1142/S0129054117500381

[18] J. Li (2016), "Reserve Output Link and FDL Together or Separately in Optical Burst Switching Networks," *Proceedings of ICTON 2016*, 1-4.
DOI : 10.1109/ICTON.2016.7550481

[19] Y. Lee. (2022). Adaptive Burst Size-based Loss Differentiation for Transmitting Massive Medical Data in Optical Internet. Journal of Digital Convergence, 20(3), 389-397.
DOI: doi.org/10.14400/JDC.2022.20.3.389

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@gmail.com