JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
https://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Le, V. H., & Vo, V. M. N. (2023). A combined delay-throughput fairness model for optical burst switched networks. *Journal of Information and Communication Technology, 22*(2), 183-205. https://doi.org/10.32890/jict2023.22.2.2

A Combined Delay-Throughput Fairness Model for Optical Burst Switched Networks

*[1]Van Hoa Le & [2]Viet Minh Nhat Vo
Hue University, Vietnam

*levanhoa@hueuni.edu.vn
vvmnhat@hueuni.edu.vn
*Corresponding author

Received: 5/7/2022 Revised: 28/12/2022 Accepted: 1/1/2023 Published: 3/4/2023

ABSTRACT

Fairness is an important feature of communication networks. It is the distribution, allocation, and provision of approximately equal or equal performance parameters, such as throughput, bandwidth, loss rate, and delay. In an optical burst switched (OBS) network, fairness is considered in three aspects: distance, throughput, and delay. Studies on these three types of fairness have been conducted; however, they have usually been considered in isolation. These fairness types should be considered together to improve the communication performance of the entire OBS network. This paper proposes a combined delay-throughput fairness model, where burst assembly and bandwidth allocation are improved to achieve both delay fairness and throughput fairness at ingress OBS nodes. The delay fairness and throughput fairness indices are recommended as metrics for adjusting the assembly queue length and allocated bandwidth for priority flows. The simulation results showed that delay and throughput fairness could be achieved simultaneously, improving the overall communication performance of the entire OBS network.

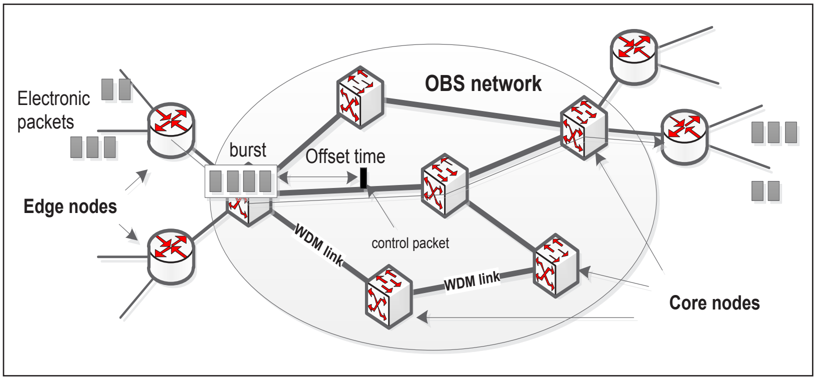
Keywords: OBS networks, delay fairness, throughput fairness, combination model, adaptive control.

INTRODUCTION

Optical burst switching (OBS) can potentially be used for next-generation optical packet-switched networks as it combines the time division multiplexing characteristics of optical packet switching and the wavelength division multiplexing characteristics of optical circuit switching. Furthermore, OBS overcomes the current limitation of optical packet switching technologies, such as optical buffers or switches at nanosecond speeds. The OBS network has thus attracted much attention from researchers over the past few decades (Zalesky, 2009). Architecturally, an OBS network consists of edge and core nodes connected together (Figure 1). The ingress node receives data (e.g., IP packets) from access networks and assembles them into bursts. At the output of the ingress node, bandwidth (wavelength channel) is allocated to carry generated bursts. At the intermediate (core) nodes, the burst is switched all-optically, and this is done until the burst reaches the egress node. Each burst is guided by a burst control packet (BCP), which reserves resources at all intermediate nodes, for an all-optical transmission.

Figure 1

An Example of a Mesh OBS Network



Routing is the activity of determining the path of a burst at the ingress node. Routing often targets multiple objectives, such as maximum burst delivery rate, minimum delay, blocking probability, and use cost (Cui & Srivastava, 2022). Establishing a path for a burst can be fixed, where a BCP reserves the resources and releases them after a burst transmission is completed (Singh et al., 2022). However, most channels are dynamically allocated and scheduled along different paths to accommodate variable available bandwidth conditions and at the same time, meet requirements for service differentiation (Kumar et al., 2022a). A modified OBS ingress node architecture has also been proposed in response to these dynamic and real-time allocation and scheduling requirements (Zeghid et al., 2021).

At the core node, resource allocation and scheduling for a burst is done by its BCP. Contention can occur during resource reservation and it is the main cause of communication performance degradation in OBS networks. A blocking probability analysis for resource reservations is often performed to assess the possibility of contention. The Markov model can be used to analyse the blocking probability for service differentiation (Barpanda et al., 2021) or optical burst switched data centres (Shaddad et al., 2021). In cases where contention is unavoidable, contention resolution techniques, such as wavelength conversion, fibre delay link (FDL) buffer usage, or deflection routing, can be invoked, where FDL is proved to be more effective than other techniques through a performance analysis comparison (Naji et al., 2022). Burst segmentation and void-filling strategies are also proposed to reduce burst loss rate due to contention (Kumar et al., 2022b). The void is the idle bandwidth generated between two consecutively scheduled bursts.

The management and control of BCPs are of great significance in improving the communication performance of the entire OBS network. The commonly used approach is to determine the best offset time, which is the time interval between a BCP and a corresponding burst so that the BCP can reserve resources at intermediate nodes before its burst arrives, and at the same time, these resources are not kept for too long (Yoo & Qiao, 2002). Other studies have exploited the fields in the BCP header. For example, the QoS field has been used to determine the burst priority if contention occurs (Sarwar et al., 2008), and the idle field has been exploited to carry the void length for adjusting the burst length (Vo et al., 2020). Nesting the offset time into the assembly time has also been suggested to reduce the buffering time at the ingress node (Sui et al., 2006; Mikoshi & Takenaka, 2008). These proposals have aimed to improve the performance metrics of OBS networks, such as fairness, load balancing, and throughput.

Fairness in OBS networks can be considered in three aspects: distance, throughput, and delay. Distance fairness is often indicated by a comparison of the burst-dropping probability between the burst with the long path and the burst with the short path. In particular, bursts with a long path have a higher dropping probability than bursts with a short path (Hsu & Yang, 2008). For an OBS network where the priority policy is based on offset time, the closer the burst is to the destination, the higher is the dropping probability compared to that of a burst that has just started. The reason is that the offset time is gradually subtracted as the burst approaches the destination. This is the second form of distance unfairness (Hsu & Yang, 2008).

Throughput fairness involves dealing with the case where the bad flow (the one that uses more bandwidth than what was allocated) takes up the bandwidth of the good flow (the one that uses less bandwidth than what was allocated) and cases a common loss when their total throughput is higher than the bandwidth capacity of the shared link. Delay fairness focuses on reducing the burst buffering delay of high-priority flows compared to low-priority flows (Sui et al., 2006). Most previous studies have suggested solutions to achieve individual fairness. Only Orawiwattanakul et al. (2010) proposed a combination of fair bandwidth allocation and distance fairness. Evidently, finding a solution that can achieve multiple fairness simultaneously can improve the communication performance of OBS networks. Nevertheless, this can also negatively impact individual fairness. A compromise can be aimed at achieving common fairness while having a negligible impact on individual fairness.

This paper proposes a combined delay-throughput fairness model, where burst assembly and bandwidth allocation are improved to achieve both delay fairness and throughput fairness. In an OBS network, the communication performance depends mainly on how well the incoming flows are controlled at the ingress edge node. This combined model was therefore implemented at ingress OBS nodes. The delay fairness and throughput fairness indices were used as metrics for adjusting the assembly queue length and allocated bandwidth for priority flows. A parameter *max_delay* was also used as a component to control the delay fairness during burst assembly and to control the throughput fairness during bandwidth allocation. The simulation results showed that delay fairness and throughput fairness can be achieved simultaneously, improving the overall communication performance of the entire OBS network.

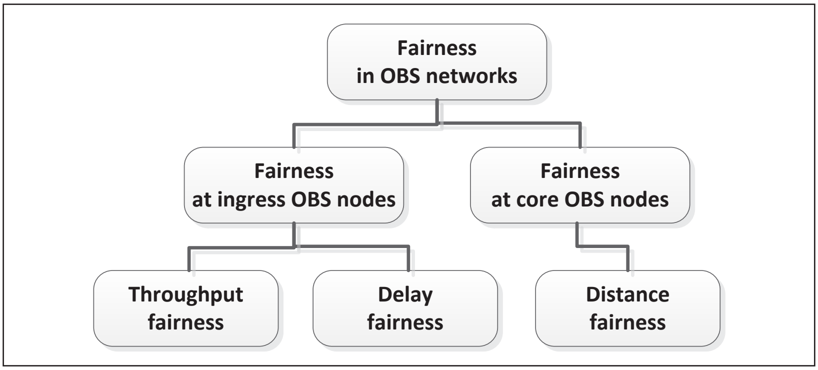
The next section summarises fairness models used in OBS networks and analyses delay fairness and throughput fairness. The proposed combined delay-throughput fairness model is presented next, wherein the two main improvements, burst assembly control and bandwidth allocation adjustment at the ingress OBS node, are examined in detail. Following this, simulations to evaluate the performance of the proposed model are presented, and the final section presents the conclusions.

RELATED WORKS

Fairness in OBS networks can be considered at the ingress node, the core node, or both the ingress and core nodes (Figure 2). Studies on the ingress node focused on two aspects: delay fairness (Sui et al., 2006; Vo et al., 2018; Wang et al. 2019) and throughput fairness (Liu et al., 2005; Orawiwattanakul et al., 2009, Le et al., 2018), while distance fairness (Hsu & Yang, 2008; Sarwar et al., 2008; Nleya et al., 2019; Nassar & Tachibana, 2020) is considered at the core node. A detailed review of these three types of fairness is presented in the following paragraphs.

Figure 2

Taxonomy of Fairness Based on Location



In delay fairness, bursts are classified according to their priority class: a high-priority burst has a shorter burst buffering time than a low-priority burst. The goal of delay fairness is to ensure that the high-priority burst is sent earlier to increase the probability of successful resource reservation. However, this interpretation reflects the essence of the fairness concept as approximate or equal distribution, allocation, and provision of performance parameters (Khan et al., 2016). Vo et al. (2018) proposed a schema of burst assembly with delay fairness (BADF), in which the concept of delay fairness was considered as the equivalent or equal ratio of end-to-end delay and the maximum delay between burst flows. Furthermore, to meet the requirement of service differentiation in terms of delay, two constraints were added: (1) A burst belonging to the high-priority class has a low end-to-end delay; and (2) the end-to-end delay of a burst is not greater than its maximum delay. Accordingly, the concept of delay fairness reported by Vo et al. (2018) included the concept of delay fairness proposed by Sui et al. (2006). Recently, delay fairness was also investigated by Wang et al. (2019) when considering priority-based assembly. An algorithm known as QoS-adaptive max burst length max assembly period was proposed to guarantee a lower assembly delay for high-priority bursts without creating additional overhead and impact on the assembly of low-priority bursts. The simulation results indicated that the assembly delay of high-priority bursts could be reduced by 2.81 percent –14.68 percent.

Throughput fairness, also known as rate fairness, refers to the fair bandwidth allocation of connections in proportion to the bandwidth provided and the available bandwidth shared by the common link (Orawiwattanakul et al., 2009). Each connection carries a data flow, and connections can share the same link or the same wavelength of a link. Therefore, without service isolation and protection mechanisms, bad flows (those with traffic that exceed the provided bandwidth) could send too much traffic to the core network. Consequently, a good stream (the one with traffic that does not exceed the provided bandwidth) will suffer from a common high probability of data loss. A simple solution frequently used is to limit and eliminate traffic that exceeds the allocated bandwidth of each connection. However, this approach is inefficient because the unused bandwidth of some connections is wasted if not used for bad flows.

Based on the max-min fairness model developed by Stoica et al. (2003), Liu et al. (2005) proposed a fair bandwidth allocation scheme such that when congestion occurs, the good connection is protected from the bad connection while still taking advantage of the available bandwidth. Nevertheless, the bandwidth cannot be fully utilised in an OBS network because there is always an inevitable gap that forms between bursts that share the same wavelength channel. Therefore, Liu et al. (2005) converted this fair bandwidth allocation into a loss probability corresponding to each connection, which is known as theoretical loss probability. The algorithm suggested by Liu et al. (2005) attempted to keep the actual loss probability fluctuating around this theoretical level to ensure fairness among connections. However, maintaining the actual loss probability close to the theoretical loss probability can lead to inbound traffic restriction and consequently the inefficient use of bandwidth.

Orawiwattanakul et al. (2009) proposed a rate fairness preemption (RFP) method to allocate bandwidth fairly to flows based on the max-min fairness criteria and handle fair congestion between bursts. Specifically, in the event of a contention, RFP allows bursts belonging to the bad flow to take over the channel from bursts belonging to the good flow. The burst belonging to bad flows is prioritised only when all wavelengths are busy, i.e., all connections are in use but still have idle bandwidth, to ensure more efficient bandwidth utilisation. Moreover, only edge nodes monitor the arrival rate of bursts, while core nodes perform RFP-based bandwidth allocation when the rate of incoming traffic changes significantly. Therefore, RFP does not increase the load of the core network. However, when implementing RFP, it is necessary to maintain two BCPs, i.e., forward BCP (FBCP) and back BCP (BBCP), to exchange information between the source and destination nodes. Consequently, the complexity of the algorithm and bandwidth for information exchange would considerably increase.

In solving this problem, Le et al. (2018) proposed a throughput-based fair bandwidth allocation (TFBA) schema, wherein the control of throughput fairness was based on two parameters: the actual bandwidth, ATi, and the allocated bandwidth according to the max-min fairness, ABi , of connection i. When a contention occurs, TFBA first considers whether the incoming burst is in the bad or good flow: if ATi > ABi, the incoming burst is in the bad flow and it is dropped to reserve resources for the burst in the good flow. Conversely, if ATi < ABi, the incoming burst belongs to a good flow, and the ratio of the actual bandwidth and allocated bandwidth between the good connection i (yi) and the bad connection j (yj) must be compared. If yi is less than yj, the contended burst is dropped; otherwise, the incoming burst is dropped. The ratios yi and yj of connections i and j, respectively, are determined by Equation 1.

$$y_i = \frac{AT_i}{AB_i} \text{ and } y_j = \frac{AT_j}{AB_j} \tag{1}$$

Distance fairness is known as the fairness problem between a flow with a long path and a flow with a short path, where a flow with a long path must go through more intermediate nodes. Thus, there is a higher probability of data loss than a flow with a short path (Hsu & Yang, 2008). Furthermore, as most OBS networks use the just-enough-time (JET) protocol (Yoo & Qiao, 2002) for signalling, the contention solution is based on an offset time comparison, where a burst with a longer offset time takes precedence over a burst with a shorter offset time. As a result, the closer the burst is to the destination, the higher the probability of being dropped because its offset time is shorter due to gradual subtraction (Sarwar et al., 2008). This is another unfair distance issue.

Recently, distance fairness was addressed by Nleya et al. (2019), where a scheme called the restricted intermediate node buffering-based routing and wavelength assignment was proposed to select between the primary and deflection paths when a contention occurred. Distance fairness was then used as a performance metric to select the best route. Distance fairness can also be considered at the output of an ingress node (Nassar & Tachibana, 2020), where the generated bursts were clustered based on their path length (hop count). The loss probability of the burst with a short path was set higher than that of a burst with a long path. The ingress node calculated this probability based on the number of received acknowledgement (ACK) and negative acknowledgement (NACK) messages and then dynamically changed the order of bursts within the burst cluster. The simulation results showed that the local fairness for each ingress node improved regardless of the amount of traffic on each link.

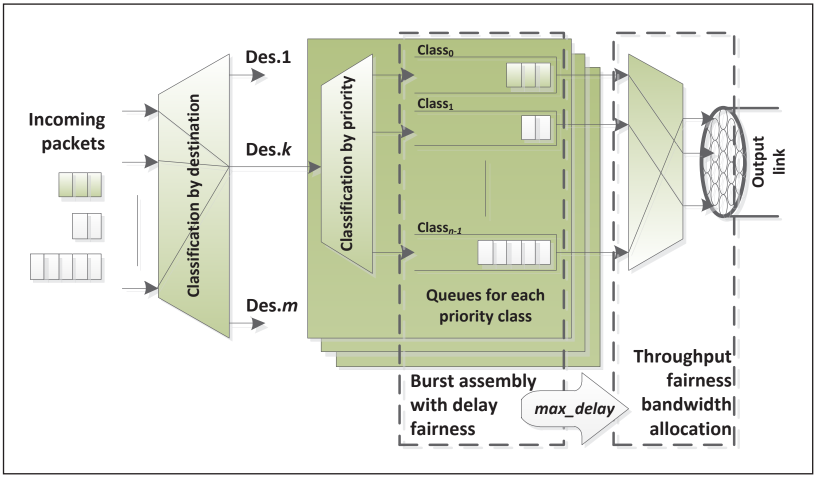
Fairness is also considered for throughput and distance simultaneously to improve communication performance. Specifically, Orawiwattanakul et al. (2010) suggested a solution, called the rate and distance fairness preemption (RDFP) schema, for fair bandwidth allocation combined with distance fairness. Their schema had three stages: (1) First, bandwidth was allocated according to max-min fairness to flows as shown by Liu et al. (2005) and RDFP protected the good connections from the bad ones. (2) RDFP then provided distance fairness only for connections with less traffic than the maximum allocated bandwidth (good connection). (3) If a connection sent more traffic than the bandwidth allocated to it (bad connection), a higher drop probability was set for the excess traffic and the distance fairness was not provided for this connection. Therefore, the simultaneous consideration of multiple fairness parameters, rather than individual ones, can improve the communication performance of the OBS network. Nevertheless, there are insufficient studies on other combinations, such as the combination of delay fairness and throughput fairness. The next section describes this combination in detail.

COMBINED DELAY-THROUGHPUT FAIRNESS MODEL

The combined delay-throughput fairness (CDTF) model was based on the idea that delay fairness needed to satisfy the maximum delay constraint of the data carried in a burst. Therefore, the delay fairness was attributed to an improvement of the BADF scheme (Vo et al., 2018), called iBADF, which used the parameter *max_delay* to regulate the time threshold $T_a(i)$. The output of the iBADF scheme was the remaining value of *max_delay*, which continued to be used to control the bandwidth allocation. The throughput fairness was based on an The combined delay-throughput fairness (CDTF) model was based on the idea that delay fairness needed to satisfy the maximum delay constraint of the data carried in a burst. Therefore, the delay fairness was attributed to an improvement of the BADF scheme (Vo et al., 2018), called iBADF, which used the parameter max_delay to regulate the time threshold Ta(i). The output of the iBADF scheme was the remaining value of max_delay, which continued to be used to control the bandwidth allocation. The throughput fairness was based on an improvement of the TFBA schema (Le et al., 2018), called iTFBA, where a throughput control mechanism was implemented by selecting bursts to drop if a contention occurred. Once again, max_delay was utilised to make the decision whether to drop the burst or not. The CDTF model was implemented at the ingress node as shown in Figure 3, where burst assembly was performed on n qeues deployed for *n* incoming data classes,$n \in \mathrm{N}^+$.The completed bursts in each queue formed a burst flow. Burst flows arriving at the output port were allocated a fair amount of bandwidth.

Figure 3

Combined Delay-Throughput Fairness Model



Improved Burst Assembly with Delay Fairness

In the BADF scheme, burst assembly was performed in two stages: (1) sending BCP early at the estimation time $T_e(i)$ and (2) completing the burst when the assembly threshold $T_a(i)$ or the estimated length $L^e(i)$ was reached (Vo et al., 2018). $T_e(i)$ is the difference of $T_a(i)$ and $T_o(i)$. Nesting the offset time $T_o(i)$ into the assembly time $T_a(i)$ only reduced the burst buffering time without creating the delay fairness. Therefore, a parameter x_i, which is the ratio of the average delay of data in queue i ($D(i)$) and the assembly threshold $T_a(i)$, was proposed to reflect the delay of data in this queue. The parameter x_i is determined by Equation 2. Based on Jain et al. (1984), the delay fairness index (*DFI*) is then determined by Equation 3.

$$x_i = \frac{D(i)}{T_a(i)} \tag{2}$$

$$DFI = \frac{\left(\sum_{i=1}^{n} \sigma_i x_i\right)^2}{n\sum_{i=1}^{n} (\sigma_i x_i)^2} \tag{3}$$

Fairness would increase as *DFI* approached 1, and equal to 1 when,

$$\sigma_1 x_1 = \sigma_2 x_2 = \ldots = \sigma_n x_n,$$

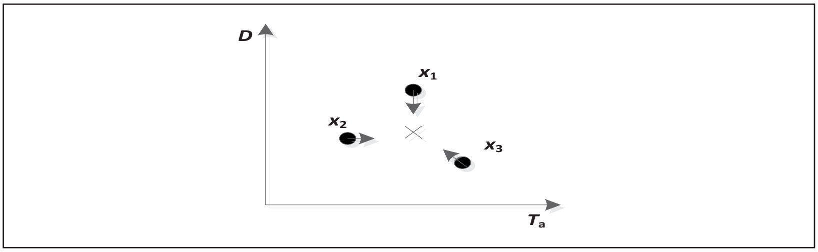
where σ_i is the weight of x_i, $0 < \sigma_i < 1$ and $\sum_{i=1}^{n} \sigma_i = 1$, and *n* is the number of queues.

The value of $D(i)$ varied depending on the rate of data that arrived at queue *i*. In pushing *DFI* to 1, the assembly threshold $T_a(i)$ was adjusted so that the points x_i became close together in space (D, T_a), as shown in the example in Figure 4. However, the adjustment of $T_a(i)$ could not exceed the maximum delay threshold of the data in queue *i*. Therefore, in the improved version proposed in this article, the value of *max_delay* was set to the minimum of maximum delay thresholds as in Equation 4, where RTT_p is the maximum delay threshold of packets *p* in queue *i*.

$$max\_delay(i) = \min(RTT_p) \tag{4}$$

Figure 4

An Example x_i of 3 Priority Burst Classes Distributed in Space (D, T_a)



The iBADF algorithm is described in Algorithm 1 as follows:

Algorithm 1: Improved Burst Assembly with Delay Fairness (iBADF)

```
Input:
  $T_a(i)$, $T_o(i)$: the assembly time and the offset time of queue $i$
  $Q_i$: list of incoming packets in queue $i$

Output:
  max_delay(i): maximum remaining delay of the burst after leaving queue $i$

While there is a packet $p$ to the queue $Q_i$ do
    If the packet arrives at the empty queue $B(i)$ then
        Enable timer $timer(i) := 0$;
        Determine the estimated window size $T_e(i)$, which is also the time to send BCP
        max_delay(i) is equal to the round trip time (RTT) of the first incoming packet
    End If
    If the maximum delay of incoming packet $p$ ($RTT_p$) is less than max_delay(i) then    Reassign max_delay(i) := RTTp
        Increase the current length of queue $L_p$ for each incoming packet $p$
    If $timer(i) \geq T_e(i)$ send BCP then
        Determine the packet arrival rate $\lambda_{cur}(i) := B(i)/T_e(i)$
        Adjust the weight $\alpha(i) := \lambda_{cur}(i) / (\lambda_{cur}(i) + \lambda_{avg}(i))$
        Determine the average packet rate $\lambda_{avg}(i) := (1 - \alpha(i)) \times \lambda_{avg}(i) + \alpha(i) \times \lambda_{cur}(i)$
        Determine estimated burst length $L_e(i) := L(i) + T_o(i) \times \lambda_{avg}(i)$
    End If
    If $((|B(i)| \in [L_e(i) - max_p(i), L_e(i)])$ or $(t(i) \geq T_a(i))$ send burst then
        Empty the queue after the assembly $B(i) := 0$
        Set the current assembly time $D(i) := timer(i)$
        Calculate the ratio $x_i := D(i)/T_a(i)$
        Determine the centre $\bar{x} := \frac{\sum_{j=1}^{n} x_j}{n}$
        If ( $D(i)/(x_i + \theta \times (\bar{x} - x_i)) \leq T_o(i)$ ) then
            Determine the assembly time $T_a(i) := T_a(i)$
```

(continued)

Algorithm 1: Improved Burst Assembly with Delay Fairness (iBADF)

```
        Else If ($T_o(i) < D(i)/(x_i + \theta \times (\bar{x} - x_i)) < max\_delay(i)$) then
            Adjust $T_a(i) := D(i)/(x_i + \theta \times (\bar{x} - x_i))$ to increase the delay fairness
        Else Determine the assembly time $T_a(i) := max\_delay(i)$
        End If
    End If
End While
Return $max\_delay(i) - D(i)$
```

The computational complexity of the iBADR algorithm mainly caame from the **While** loop. Since the complexity of the instructions in **While** loop was O(1), the complexity of iBADR was then O(N), where N is the number of packets arriving in queue Q_i. The computational complexity of iBADR algorithms is comparable to that of conventional assembly algorithms, since they follow the principle of traversing all packets in a queue Q_i.

Improved Throughput Fairness Bandwidth Allocation

The throughput fairness bandwidth allocation scheme in this article was an improvement of TFBA in Le et al. (2018), in which the parameter *max_delay* was transmitted to the output of the ingress node to participate in throughput fairness control. Specifically, bursts generated from the same queue *i* formed the burst flow *i*. Differentiated burst flows were equally allocated through the TFBA scheme (Le et al., 2018). When congestion occurred, the bandwidth allocation could be adjusted. An improvement added to TFBA was a new burst dropping strategy based on the maximum remaining delay of the burst, in which the burst with low *max_delay*(*i*) received scheduling priority while the burst with high *max_delay*(*i*) was dropped. This approach not only fairly regulated the throughput of flows entering the core network, but also ensured the end-to-end delay of bursts. In measuring the fairness of each flow, the throughput fairness index (*TFI*), which was also based on Jain et al. (1984), is determined as in Equation 5.

$$TFI = \frac{\left(\sum_{i=1}^{n} \sigma_i y_i\right)^2}{n \sum_{i=1}^{n} (\sigma_i y_i)^2} \tag{5}$$

Fairness would increase as *TFI* approached 1, and equal to 1 when ,

$$\sigma_1 x_1 = \sigma_2 x_2 = \ldots = \sigma_n x_n ,$$

where σ_i is the weight of x_i, $0 < \sigma_i < 1$ and , and n is the number of queues.

The iTFBA algorithm is described in Algorithm 2 as follows:

Algorithm 2: Improved Throughput Fairness Bandwidth Allocation (iTFBA)

```
Input:
  $b(i)$: incoming burst $i$
  $\lambda_{pre}(i)$: previous throughput of flow $i$
  $max\_delay(i)$: maximum delay of burst $b_i$
  $Bw$: bandwidth of the output link
Output:
  $\lambda_{cur}(i)$: current throughput of flow $i$
Initiate the parameters: $\omega := 0.7$, $C := 1.0$, $th := 0.1$
If $b(i)$ cannot be scheduled then
    Determine the actual throughput in each estimation window $\lambda_{cur}(i) := p_w(i)/T_w(i)$
    If there is a change in the arrival speed ($|\lambda_{cur}(i) - \lambda_{pre}(i)| > th$) then
        Determine the parameters $S := \varnothing$, $m := N$, where $N$ is the number of priority classes
        Calculate the actual bandwidth capacity $U := \omega \times C$
        Repeat
            Determine $FS := U / m$, $m_{prev} := m$
            Calculate the fair ratio of connections, $F(l) := \min\{\lambda(l), FS\}$, $l \notin S$
            Determine the set of connections belonging to the good flow $S := \{j: \lambda(j) \leq FS\}$
            Determine the excess bandwidth $U := U - \Sigma_{j \in S}\lambda(j)$
            Determine the number of connections sharing excess bandwidth $m := m - |S|$
```

(continued)

Algorithm 2: Improved Throughput Fairness Bandwidth Allocation (iTFBA)

```
        Until ($m = m_{prev}$ or $m = 0$)
    End If
    Determine the bandwidth allocated for connection i, $AB(i) := F(i) \times Bw$
    If flow i is bad one ($\lambda_{cur}(i) > AB(i)$) then
        Reduce the actual throughput $\lambda_{cur}(i) := (p_w(i) - |b(i)|)/T_w(i)$
    Else If determine which burst to drop ($max\_delay(i) < max\_delay(j)$) then
        Determine the throughput of flow j, $\lambda_{cur}(j) := (p_w(j) - |b(j)|)/T_w(j)$
    Else Determine the throughput of flow i, $\lambda_{cur}(i) := (p_w(i) - |b(i)|)/T_w(i)$
    End If
Return Throughput of flow i, $\lambda_{cur}(i)$
```

The computational complexity of the iTFBA algorithm mainly came from the process of bandwidth allocation (**Repeat – Until** loop). In the **Repeat - Until** loop, there were three other loops: the first loop was to distribute the fair ratio of connections $F(l)$, the second loop was to determine the set of connections of a good flow, and the third loop calculated the redundant bandwidth. However, these three loops were independent of each other and had a complexity of $O(N)$, where N is the number of connections. In the **Repeat - Until** loop, the worst case was repeated N times, so the computational complexity of the algorithm was $O(N^2)$.

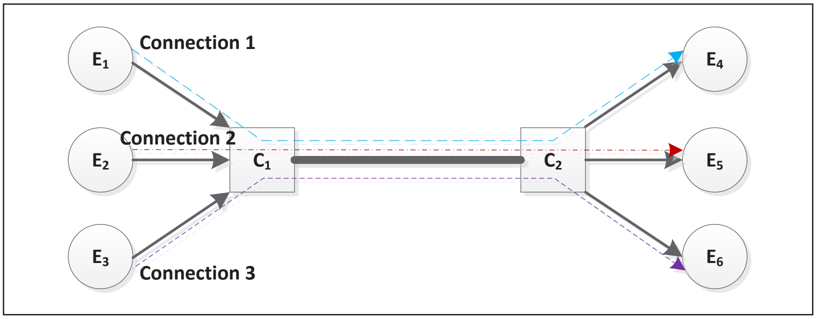
SIMULATION AND ANALYSIS

The CDTF model was implemented on Network Simulator 2 (NS2) (Information Sciences Institute, 2022) with package obs-0.9a. Incoming data (e.g., IP packets) were assumed to belong to three priority classes ($n = 3$); thus, there were three queues used for burst assembly at the ingress node. The arrival of IP packets at each queue had a Poisson distribution and their size was distributed in [500, 1,000] bytes. Assume that the maximum delay threshold (in milliseconds) of incoming packets was distributed in [0.4, 1.0]. Therefore, packets with the maximum delay in [0.4, 0.6] were put in the first queue (highest priority), packets with the maximum delay in [0.6, 0.8] were placed in the second (medium priority) queue, and the remaining packets with maximum delay in [0.8, 1.0] were entered in the third queue (lowest priority). Offset times of 0.3, 0.2, and 0.1 (in milliseconds) were assigned to queues 1, 2, and 3, respectively.

The simulation parameters are as follows. There were $W = 8$ wavelengths per outgoing link and the bandwidth of each link was 10 Gbps. The actual bandwidth utilisation of each link was $\omega = 0.7$. The simulation was performed in 1.0 s, where from 0 to 0.5 s, the incoming loads of the three connections 1, 2, and 3 were all 0.2 s (the case where the total load did not exceed the link capacity); from 0.6 s to 1.0 s, the load of connection 3 increased to 0.6 s, while the loads of connections 1 and 2 remained unchanged (in case the total load exceeded the link capacity). Since the simulation objectives only considered the performance of the connections that shared the same outgoing link, it was sufficient to use a Dumbell network as shown in Figure 5 for the implementation.

Figure 5

Dumbell Network for Simulation



The performance metrics included the byte loss rate, which was measured by the division of the number of lost bytes and the total number of sent bytes, the average end-to-end delay, and the delay-throughput fairness index. Based on the proposal of Jain et al. (1984), the delay-throughput fairness index (*DTFI*) was determined by

Equation 6, where δ is a correlation coefficient between delay and throughput fairness. Depending on the weight of delay or throughput, the correlation coefficient was set to a specific value. If the weights of delay or throughput were the same, $\delta = 0.5$.

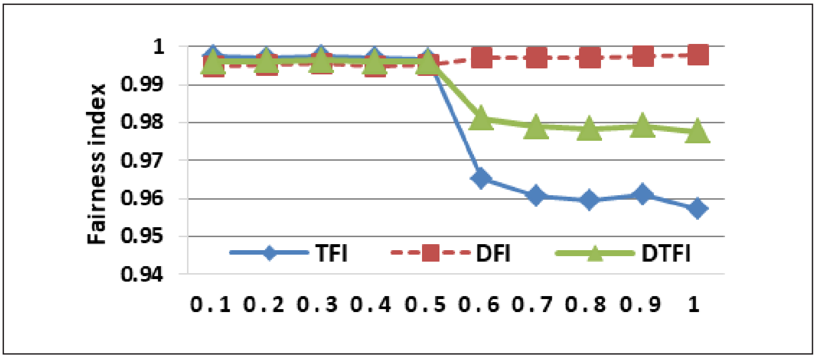
$$DTFI = \frac{\left(\sum_{i=1}^{n}(\delta x_i + (1-\delta)y_i)\right)^2}{n\sum_{i=1}^{n}(\delta x_i + (1-\delta)y_i)^2} \tag{6}$$

Comparison of the Delay, Throughput, and Delay-Throughput Fairness

When comparing fairness, in terms of the fairness index, *DTFI* was in between *DFI* and *TFI* as shown in Figure 6. This was clearly a compromise to achieve two fairness at the same time. Combining iBADF and iTFBA helped CDTF improve the fairness throughput yet reduced delay fairness. Increasing throughput fairness through bandwidth allocation adjustment and selective burst drop resulted in an increase in the average end-to-end delay. However, the increase in the *TFI* side was equivalent to the decrease in the *DFI* side. This is the case where delay fairness and throughput fairness had the same weight (δ = 0.5). If there was a preference for delay fairness or throughput fairness, *DTFI* would shift accordingly towards delay fairness (up) or throughput fairness (down). Nevertheless, it would always represent a compromise between *DFI* and *TFI*.

Figure 6

Comparison of the Fairness Indexes: TFI, DFI, and DTFI

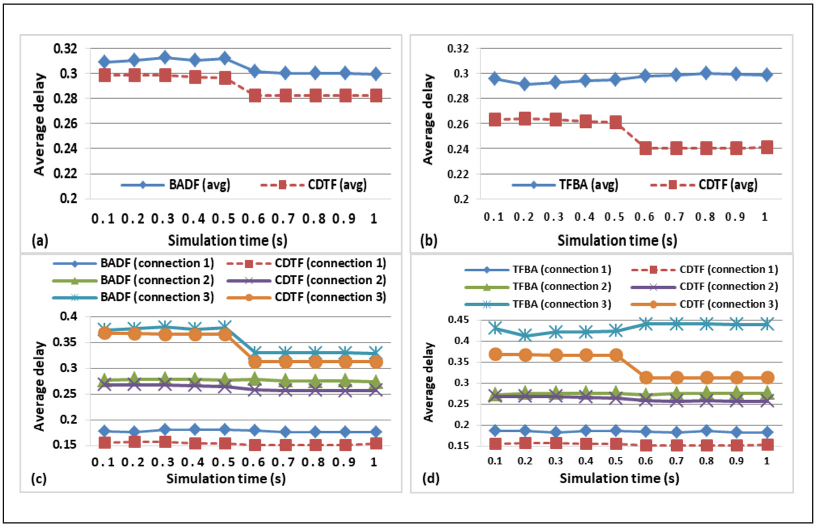


Comparison of the Average End-to-End Delay

When compared with BADF and TFBA in terms of the average end-to-end delay, CDTF was always better as shown in Figure 7. Since TFBA did not implement delay reduction at the ingress node (during burst assembly), it had a higher delay than CDTF. However, with BADF, the average delay of CDTF was still lower thanks to the improved mechanism of burst dropping control, in which a burst with low maximum delay (*max_delay*) received priority scheduling, while a burst with higher *max_delay* was dropped. This feature helped CDTF to improve in average delay when compared to BADF. Figure 7(c, d) shows a comparison of the average delay of each connection, where CDTF always achieved the lowest delay for all three connections, i.e., connections 1, 2, and 3.

Figure 7

Comparison of the Average Delay between BADF, TFBA, and CDTF



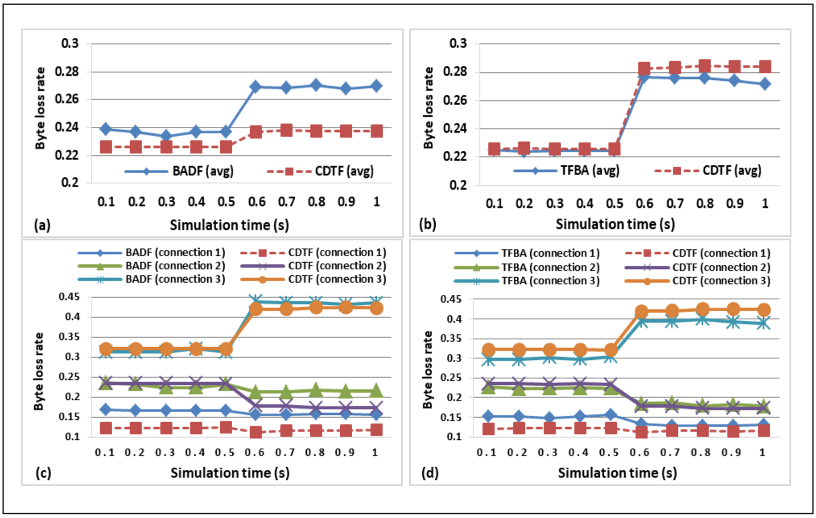
Comparison of the Byte Loss Rate

When compared with BADF (Figure 8a), the byte loss rate of CDTF was significantly lower (about 4.5% in the period [0, 0.5 s] and 11.7 percent in the period [0.5, 1.0 s]). In the BADF scheme, throughput fairness was ignored, so there was no bandwidth allocation adjustment as incoming traffic increased. This caused a high loss for

BADF compared to CDTF. In contrast with TFBA (Figure 8b), the loss rate of CDTF was slightly higher (about 0.6% in the period [0, 0.5 s] and 3.2 percent in the period [0.5, 1.0 s]). In fact, the TFBA scheme used the traditional hybrid burst assembly algorithm, where the time threshold and the length threshold were fixed. However in CDTF, to ensure delay fairness, the burst assembly threshold $T_a(i)$ was dynamically adjusted, which affected the rate of generated burst flow. This impacted the bandwidth allocation adjustment at the output port and caused an increase in the loss rate, although not significantly. Figure 8(c,d) shows a comparison of the byte loss rate of each connection, where CDTF always achieved the lowest loss rate for good connection (connections 1 and 2) and slightly higher loss rate for bad connection (connection 3).

Figure 8

Comparison of the byte loss rates between BADF, TFBA, and CDTF



CONCLUSION

The paper proposed a model that combined delay fairness and throughput fairness. This model is called the combined delay-throughput fairness (CDTF) model. It implements delay fairness through burst assembly and throughput fairness through dynamic bandwidth allocation. A parameter *max_delay* is used to control dynamic bandwidth allocation and selective burst drops when contention occurs. The simulation results indicated that the CDTF model performed considerably better compared to other models in terms of byte loss rate and average end-to-end delay. However, achieving combined delay-throughput fairness also negatively impacted each individual fairness, resulting in higher delay fairness and lower throughput fairness. This implies that more research is needed to improve combined fairness in a way that does not impact individual fairness. The combination of delay, throughput, and distance fairness should also be considered.

ACKNOWLEDGMENT

This work was supported/partially supported by Hue University under the Core Research Program, Grant No. NCM.DHH.2019.05.

REFERENCES

Barpanda, R. S., Turuk, A. K., & Sahoo, B. (2021). Analysis of wavelength reservation based quality of service differentiation in optical burst switching networks using Markov model. *Egyptian Informatics Journal*, *22*(3), 363–372. https://doi.org/10.1016/j.eij.2016.06.006

Cui, L. S., & Srivastava, G. (2022). QoS routing algorithm for obs networks based on a multi-objective genetic algorithm. *IEEE Access*, *10*, 12047–12056. https://doi.org/10.1109/ACCESS.2021.3138380

Hsu, C. F., & Yang, L. C. (2008). On the fairness improvement of channel scheduling in optical burst-switched networks. *Photonic Networks Communication*, *15*(1), 51–66. https://doi.org/10.1007/s11107-007-0087-9

Jain, R., Chiu D. M., & Hawe, W. R. (1984). A quantitative measure of fairness and discrimination for resource allocation in shared computer system. *DEC technical report TR301 1984, cs.NI/9809 (DEC-TR-301) 1984* (pp. 1–38). http://www.cs.wustl.edu/~jain/papers/ftp/fairness.pdf

Khan, U. U., Dilshad N., Rehmani M. H., & Umer, T. (2016). Fairness in cognitive radio networks: Models, measurement methods, applications, and future research directions. *Journal of Network and Computer Applications*, *73,* 12–26. https://doi.org/10.1016/j.jnca.2016.07.008

Kumar, V. K. A., Kumar, M. R., Shribala, N., Singh, N., Gunjan, V. K., Siddiquee, K. N. E. A., & Arif, M. (2022a). Dynamic wavelength scheduling by multiobjectives in OBS networks. *Journal of Mathematics*, *2022*. https://doi.org/10.1155/2022/3806018

Kumar, V. K. A., Reddy, K. S., Prashanth, A., & Kumar, B. R. (2022b, September). Robust strategy for optical burst scheduling in WDM networks through void filling and segmentation. In *Proceedings of International Conference on Communication and Computational Technologies* (pp. 115–129). Springer. https://doi.org/10.1007/978-981-19-3951-8_10

Le, V. H., Vo, V. M. N., & Le, M. T. (2018). Throughput-based fair bandwidth allocation in OBS networks. *ETRI Journal*, *40*(5), 624–633. https://doi.org/10.4218/etrij.2017-0253

Liu, Y., Chua K. C., & Mohan, G. (2005). Achieving max-min fairness in WDM optical burst switching networks. *HPSR. 2005 workshop on high performance switching and routing* (pp. 187–191). https://doi.org/10.1109/HPSR.2005.1503220

Mikoshi, T., & Takenaka, T. (2008, April). Improvement of burst transmission delay using offset time for burst assembly in optical burt switching. In *7th Asia-Pacific Symposium on Information and Telecommunication Technologies 2008* (pp. 13–18). IEEE. https://doi.org/10.1109/APSITT.2008.4653532

Naji, L. A. W. A., Eltahir, I. K., Al-Shargabi, M., & Haider, A. S. (2022). Performance analysis comparison of optical burst switching networks' contention resolution techniques. *Indonesian Journal of Electrical Engineering and Computer Science*, *25*(3), 1539–1548. https://doi.org/10.11591/ijeecs.v25.i3.pp1539-1548

Nassar, B. O., & Tachibana, T. (2020). Adaptive burst reordering for burst-cluster transmission to improve fairness in OBS networks. *Journal of High Speed Networks*, *26*(1), 1–12. https://doi.org/10.3233/JHS-200625

Nleya, B., Khumalo, P., & Mutsvangwa, A. (2019, August). A restricted intermediate node buffering-based contention control scheme for OBS networks. In *International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)* (pp. 1–6). IEEE. https://doi.org/10.1109/ICABCD.2019.8851001

Information Sciences Institute. (2022). *The network simulator – ns-2*. Information Sciences Institute. https://www.isi.edu/nsnam/ns

Orawiwattanakul, T., Ji, Y., Zhang, Y., & Li, J. (2009). Fair bandwidth allocation in optical burst switching networks.

Photonic Networks Communication, *15*(1), 51–66. https://doi.org/10.1007/s11107-007-0087-9

Orawiwattanakul, T., Ji, Y., Zhang Y., & Li, J. (2010, December). Fair bandwidth allocation with distance fairness provisioning in optical burst switching networks. In *IEEE Global Telecommunications Conference GLOBECOM 2010* (pp. 1–5). IEEE. https://doi.org/10.1109/GLOCOM.2010.5683193

Sarwar, S., Wallentin, L., Franzl, G., & Van As, H. R. (2008, September). Composite burst assembly with high-priority packets in the middle of burst. In *5th International Conference on Broadband Communications, Networks and Systems 2008* (pp. 140–145). IEEE. https://doi.org/10.1109/BROADNETS.2008.4769058

Shaddad, R. Q., Al-Ssarary, A. M., Al-Mekhlafi, S. A., Qaid, M. D., & Farhan, Z. M. (2021, March). Contention resolution of optical burst switching for data center. In *2021 International Conference of Technology, Science and Administration (ICTSA)* (pp. 1–4). IEEE. https://doi.org/10.1109/ICTSA52017.2021.9406552

Singh, S., Kaur, D., & Verma, S. (2022, November). A circuit switching-based resource reservation approach for optical burst switched networks. In *Proceedings of Third Doctoral Symposium on Computational Intelligence* (pp. 685–699). Springer. https://doi.org/10.1007/978-981-19-3148-2_59

Stoica, I., Shenker S., & Zhang, H. (2003). Core-stateless fair queuing: A scalable architecture to approximate fair bandwidth allocations in high-speed networks. *IEEE/ACM Transaction Networks, 11*(1), 33–46. https://doi.org/10.1109/TNET.2002.808414

Sui, Z., Zeng, Q., & Xiao, S. (2006). Adaptive assembly on delay fairness in optical burst switched networks. *Journal Optical Communication*, *27*(1), 35–38. https://doi.org/10.1515/JOC.2006.27.1.35

Vo, V. M. N., Le, V. H., & Le, M. T. (2018). Delay fairness using the burst assembly for service differentiation. *ETRI Journal*, *40*(3), 347–354. https://doi.org/10.4218/etrij.2017-0077

Vo, V. M. N, Pham, T. D., Dang, T. C., & Le, V. H. (2020). A mechanism of QoS differentiation based on offset time and adjusted burst length in OBS networks. *Turkish Journal of Electrical Engineering and Computer Sciences*, *28*(5), 2808–2820. https://doi.org/10.3906/ELK-1906-87

Wang, Y., Chen, T., & Zhou, N. (2019, June). Space-based optical burst switching assembly algorithm based on QoS adaption. In *IEEE 11th International Conference on Communication Software and Networks (ICCSN)* (pp. 101–105). IEEE. https://doi.org/10.1109/ICCSN.2019.8905264

Yoo, M., & Qiao, C. (2002). Just-enough-time (JET): A high speed protocol for bursty traffic in optical networks. In *Digest of the IEEE/LEOS Summer Topical Meeting: Vertical-Cavity Lasers/Technologies for a Global Information Infrastructure/WDM Components Technology/Advanced Semiconductor Lasers and Applications/Gallium Nitride Materials, Processing, and Devices 1997* (pp. 26–27). IEEE. https://doi.org/10.1109/LEOSST.1997.619129

Zalesky, A. (2009). To burst or circuit switch ? *IEEE/ACM Transaction Networks*, *17*(1), 305–318. https://doi.org/10.1109/TNET.2008.923718

Zeghid, M., Amaseb, K., Ahmed, H. Y., Khan, A. N., Chehri, A., Sharma, T., & Nisar, K. S. (2021). Modified optical burst switching (OBS) based edge node architecture using real-time scheduling techniques. *IEEE Access*, *9*, 167305–167321. https://doi.org/10.1109/ACCESS.2021.3132578