Path Setup Algorithms for Uniform/Nonuniform Traffic in Unidirectional Optical Compression TDM Rings

Kazuhiro Gokyu[a], Ken-ichi Baba[b], and Masayuki Murata[b]

[a]Department of Informatics and Mathematical Science, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka 560–8531, Japan
[b]Cybermedia Center, Osaka University
5–1 Mihogaoka, Ibaraki, Osaka 567–0047, Japan

ABSTRACT

In this paper, we propose path accommodation methods for unidirectional rings based on an optical compression TDM (OCTDM) technology. We first derive a theoretical lower bound on the numbers of slots and frames, in order to allocate all paths among nodes. Three path accommodation algorithms for the all–optical access are next proposed to achieve the lower bound as close as possible. Path splitting is next considered to improve the traffic accommodation. Finally, we analyze the packet delay time for given numbers of slots/frames, which are decided by our proposed algorithms. Numerical examples are also shown to examine the effectiveness of our proposed algorithms including path accommodation and path–splitting methods.

Keywords: Optical Compression TDM, Path Accommodation Method, Optical Unidirectional Ring Network, Theoretical Lower Bound

1. INTRODUCTION

A packet–switched ring with all–optical access can be realized by optical wavelength multiplexing (WDM) or optical time–division multiplexing (OTDM) techniques. In a last few years, it becomes evident that an optical pulse compression/expansion technology [1–3] is useful for the OTDM rings, which is called OCTDM (Optical Compression TDM). OCTDM can provide high–speed backbone networks with one to tens of Gbps [4,5]. As described in [4], as the optical node of the OCTDM ring receives the packet from LAN, bit intervals are shortened to fit the time–slot length of the backbone ring. When receiving the packet at the destination node, it is lengthened to fit the LAN speed.

In OCTDM, we need some routing policy to decide how each slot within a frame is used by every node pair. In the conventional TDM, it is easy to accommodate the traffic on the ring as the following way. Suppose that the ring has N nodes, numbered from 0 to $N-1$. The ith slot within the frame (consisting of N slots) is allocated to the ith source node. The ith source node always transmits the packet on the ith slot. The destination node can retrieve the packet by observing the destination address in the header. It implies that the destination node can receive at most $N-1$ packets within the frame time. In OCTDM, on the contrary, the number of slots transmitted (and received) within the frame should be limited by the number of transceivers since it employs the optical pulse compression/expansion for ring access [6]. The path accommodation methods suitable to OCTDM with bidirectional rings is shown in [7].

In this paper, we first propose the path accommodation methods for unidirectional rings with OCTDM. The path–splitting method is next investigated to improve the degree of path accommodation. We had not considered it with bidirectional rings in [7]. An ideal realization of optical networks is achieved by all–optical connection between every node pair. However the performance of OCTDM rings can be actually improved by carefully splitting several paths at intermediate nodes unless an OE/EO conversion is not a bottleneck. See Section 5.4. In this paper, we first describe the path accommodation methods for all–optical access, by which we try to obtain the theoretical lower bound in the number of slots/frames. If it is unable, we allow some all–optical paths to be split in order to achieve higher performance. A similar idea of path–splitting is presented in [8] for WDM rings.

As a related work, the path allocation method for the WDM ring is shown in [9,10]. In [9], a cost effective design method is proposed for accommodating a *wavelength path* for every node pair. In their method, the number of wavelengths is a limited resource. In their companion paper [11], the time needed to accommodate all paths for a given number of wavelengths is also

Correspondence: K.G. Further author information: K.G.: E-mail: gokyuu@ics.es.osaka-u.ac.jp, K.B.: E-mail: baba@cmc.osaka-u.ac.jp, M.M.: E-mail: murata@cmc.osaka-u.ac.jp

obtained. They consider the fixed packet length, and therefore, the time is slotted in the WDM system. Thus, their system becomes similar to our OCTDM ring. However, they do not obtain the packet transmission time, which will be presented in the current paper. Also, path–splitting is not considered in [9–11].

The rest of the paper is organized as follows. In Section 2, we first describe a OCTDM ring structure and our model. In Section 3, we derive the theoretical lower bound on the number of frames necessary to accommodate all paths for given parameters (the numbers of transmitters/receivers and time–slots). In Section 4, three path accommodation algorithms are considered, and we propose a traffic splitting access method suitable to the OCTDM ring. The effectiveness of those algorithms is then compared based on the theoretical lower bounds shown in Section 3. In Section 5, we analytically obtain and evaluate the packet delay time. Conclusions and future works are summarized in Section 6.

2. THE OPTICAL PULSE COMPRESSION/EXPANSION TECHNIQUE AND THE STRUCTURES OF THE OCTDM RING

2.1. Optical Pulse Compression/Expansion Technique

An optical pulse compression/expansion technology is promising to realize the very high-speed backbone ring [6]. When the packet is put on the optical line, a bit interval is compressed by using the fiber delay loop (Fig. 1). Since the compression rate with one loop is limited, high compression rate can be achieved by using several steps if it cannot be realized at a time. Also, the compression/expansion frequency at each compression/expansion device is limited; consecutive packets compressions/expansions need some optical compression/expansion devices at each access point. A semiconductor optical amplifier (SOA) and the switch (SW) are inserted on the loop to compensate the loss on the fiber delay loop. Then, the packet is transmitted onto the ring. When the packet is received from the optical line to LAN, bit expansion is performed as a reverse procedure of bit compression. More details of the optical pulse compression technique are described in [1–3].

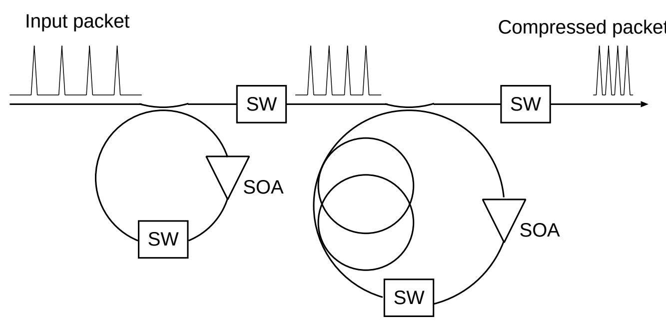


Figure 1. Bit Compression Device

2.2. Structure and Access Method of the OCTDM Ring

For explaining the structure of the OCTDM ring, we first introduce some notations. We consider an unidirectional ring with capacity B_R[bps]. It has N nodes on the ring, being time slotted. Each frame is fixed–time length having K slots. The nodes are numbered clockwise from 0 to $N-1$. Node i and Node $i+1$ are connected by link i. See also Fig. 2.

Figure 3 shows the access method to the ring at each node. Packets arriving at the node from LAN are first queued at electronic buffers. The separate buffers are prepared according to the destination node. The arriving packet is divided into mini–packets to be fitted into one time–slot. After the mini–packet is optically compressed by the fiber delay loop, it is put on the slot, which is allocated to the source/destination node pair in advance. When receiving the packet (consisting of several mini–packets) from the ring, the packet is reconstructed and is forwarded to the destination LAN.

The transmission speed of the mini–packet is identical to the backbone OCTDM ring. However, there is a limit on the number of mini–packets that the transmitter with a bit compression device can put on the slot due to speed mismatching between backbone ring and LANs. For example, when 622 Mbps LAN is connected to 40 Gbps (622 Mbps $\times$ 64) backbone ring, the node needs some optical bit compression devices by which each mini–packet is compressed into 64 times bit rate. While processing such a compression, each transmitter with the device cannot access to the backbone. Therefore, we consider that each transmitter can put only one mini–packet during the fixed time duration, which is referred to as *frame* (Fig. 4) in

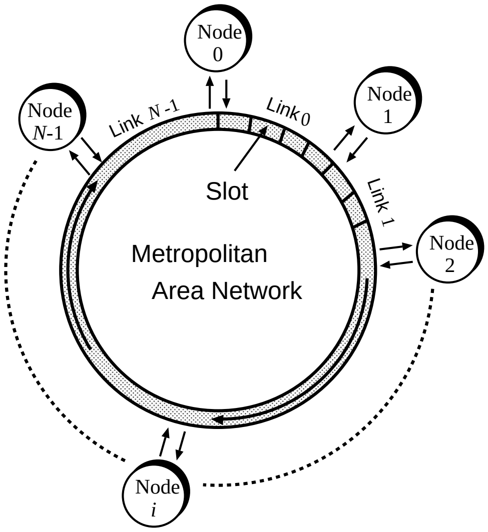


Figure 2. An Unidirectional Ring with Optical Compression TDM

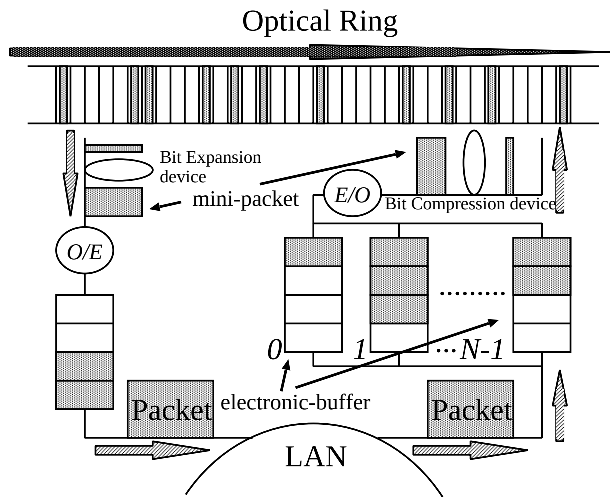


Figure 3. Node Structure and Access Method

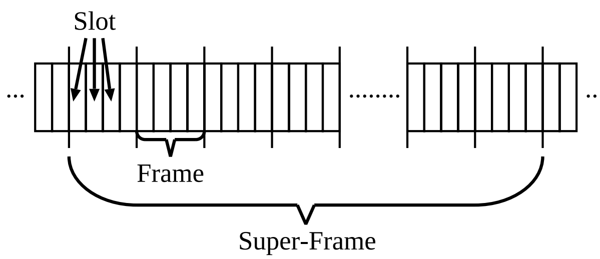


Figure 4. Relations among Slot, Frame, and Super-frame

this paper. The number of mini–packets that each receiver can receive within the frame is also limited to one. So, the length of the frame should be define by each transmitter/receiver's compression/expansion ratio. The required number of slots to accommodate all the paths between every source–destination pair is called as *super–frame* (Fig. 4), and the super–frame may consist of several frames. On the ring, the super–frame is cyclically carried.

If some node has a multiple number of transmitters (receivers), it can transmit (receive) more than one mini–packets within the frame, which possible leads to the shorter length of the super-frames. However, transmitters/receivers are costly, and those are limited resources in the OCTDM ring.

In this paper, we first focus on the optimal path accommodation method to minimize the numbers of frames/slots in the next section.

3. LOWER BOUNDS OF SUPER–FRAME LENGTH

3.1. Introduction of Notations

We assume that node i is equipped with the numbers T_i of transmitters and R_i of receivers. Let $\mathcal{T} = \{T_0, T_1, \cdots, T_{N-1}\}$ and $\mathcal{R} = \{R_0, R_1, \cdots, R_{N-1}\}$ be sets of the transmitters and receivers, respectively. We use the notation $T = j$ if every node on the ring has an identical number j of transmitters; i.e., $\mathcal{T} = \{j, j, \cdots, j\}$. Similarly, $R = k$ shows that every node can receive k mini-packets within a frame.

The traffic streams between every node pair are allocated to some slots within a super-frame according to the real traffic volume (referred to [12–14]). In this paper, each traffic stream constructing their slots is called a "path".

The path from source node i to destination node $(i+s)$ is represented by (i, s), where s is a clockwise distance between two nodes in hop counts. To make representation simple in the following presentation, we will allow to use k $(\geq N)$ in representing the node number. In that case, k should be read as $mod(k, N)$.

Our main purpose of this section is to determine the number of slots for each source/destination pair within a super–frame. Since we treat the accommodation methods of slots within the frame, we want to represent the traffic demand matrix in the unit of slots. In this and next sections, we assume that the traffic load is expressed in integer values, i.e., the required number of slots for path (i, s) takes an integer value $c^{(i,s)}$. A N by N matrix $C = \{c^{(i,s)}\}$ is given as the traffic load matrix. Hereafter, we implicitly assume that the total sum of the traffic load does not exceed the backbone ring capacity, so that it is always possible to accommodate all of paths.

3.2. Theoretical Lower Bounds

In this subsection, we derive the theoretical lower bound of the super–frame length for given N (the number of nodes), $\mathcal{T}$ (a set of the numbers of transmitters), $\mathcal{R}$ (a set of the numbers of receivers), K (the number of time–slots within the frame), and C (a traffic load matrix). We define it as $LB(N, \mathcal{T}, \mathcal{R}, K, C)$. Note that the theoretical lower bound in WDM rings was investigated in [11] under the conditions (1) that the numbers of transmitters and receivers provided by all nodes are identical, and (2) that the traffic load is uniform. The case of the bidirectional OCTDM rings is treated in [7]. We extend those methods for our unidirectional OCTDM ring in the below.

(A) The case where every node on the ring has K transmitters/receivers, and K is finite

We first consider the case where the numbers of transmitters/receivers at every node are K respectively, it means the numbers are left out of consideration, i.e, each node can access to every slot on the ring, but time–slots K is finite. We denote the total number of paths on link i by $n^{(i)}$, which can be determined from the traffic load matrix C as:

$$n^{(i)} = \sum_{j=i+2}^{i+N} \sum_{s=(i+N+1)-j}^{N-1} c^{(j,s)}. \tag{1}$$

Since each frame has K slots, the number K of paths can be set up in each frame on link i. It requires $\left\lceil \frac{n^{(i)}}{K} \right\rceil$ frames to allocate all paths on link i. The theoretical lower bound of the super–frame length, $LB(N, -, -, K, C)$, is thus given as:

$$LB(N, -, -, K, C) = \max_{0 \leq i \leq N-1} \left\lceil \frac{n^{(i)}}{K} \right\rceil. \tag{2}$$

(B) The case where K is infinite, but the numbers of transmitters/receivers at every node are finite

In this case, the total number of paths from sender node i to the other receiver nodes is given by:

$$s_p{}^{(i)} = \sum_{s=1}^{N-1} c^{(i,s)}. \tag{3}$$

Similarly, the total number of paths from sender nodes (except node i) to the receiver node i is given by:

$$r_p{}^{(i)} = \sum_{k=0}^{N-1} c^{(k,i-k)}. \tag{4}$$

Since the number of slots in each frame is infinite, the number of paths allocated for node i is bounded by the numbers of transmitters (T_i) and receivers (R_i). That is, $LB(N, \mathcal{T}, \infty, \infty, C)$ and $LB(N, \infty, \mathcal{R}, \infty, C)$ are derived as;

$$LB(N, \mathcal{T}, \infty, \infty, C) = \max_{0 \leq i \leq N-1} \left\lceil \frac{s_p^{(i)}}{T_i} \right\rceil, \tag{5}$$

$$LB(N, \infty, \mathcal{R}, \infty, C) = \max_{0 \leq i \leq N-1} \left\lceil \frac{r_p^{(i)}}{R_i} \right\rceil. \tag{6}$$

From the above two cases (A) and (B), we can determine $LB(N, \mathcal{T}, \mathcal{R}, K, C)$ using Eqs. (2), (5) and (6) as follows;

$$LB(N, \mathcal{T}, \mathcal{R}, K, C) = \max_{0 \le i \le N-1} \left(\left\lceil \frac{n^{(i)}}{K} \right\rceil, \left\lceil \frac{s_p^{(i)}}{T_i} \right\rceil, \left\lceil \frac{r_p^{(i)}}{R_i} \right\rceil \right). \tag{7}$$

From Eq. (7), we can observe that the length of the super–frame can become smaller if terms in Eq. (7) are uniformly distributed for given numbers of transmitters/receivers and the number of time–slots in the frame. Then we have the ring with higher throughput.

4. PATH ACCOMMODATION ALGORITHMS

In this section, we propose three path accommodation methods. Each of those path accommodation algorithms that we propose decides an allocation order of paths within frames. The lower bound developed in the previous section could be achieved if the algorithm works well and transmitters and receivers are effectively used. In what follows, we will first present the path accommodation method in the case with all–optical access case in Subsection 4.1. The case with path–splitting access is then treated in Subsection 4.2, where some paths are splitted at some node between source/destination nodes, in order to achieve the shorter super–frame. Subsection 4.3 is devoted to present the numerical examples.

4.1. All–Optical Access

To achieve the lower bound, we will first describe the algorithm A1, where the longest path is always examined in path allocation. In the algorithm A2, the weights of links and the number of transceivers are taken into account. The algorithm A3 reflects the accommodation balance for the effective use of the slots.

4.1.1. Algorithm A1

We first show the algorithm A1, which attempts to assign slots to the longest path first. It is simple in the sense that it does not consider the traffic load condition on every link and node. The algorithm A1 first finds the source/destination pairs requesting the path with longest distance (i.e., $s = N - 1$). For those paths, the slot is assigned from source node 0 to $(N - 1)$ if the source/destination pair requests such a path. The transmitter for the source node and the receiver for the destination node are assigned to accept those paths. Then, next longest paths with distance $s = N - 2$ are assigned. All paths are examined until paths with distance 1 are assigned slots.

The algorithm is summarized in the below. The attempt to set up the path first checks to see if transmitters, receivers, and slots are available on path (i, s) when $c^{(i,s)} \ge 1$ (see Line 7:). If it is true, path (i, s) is actually set. Then, $c^{(i,s)}$ is decremented by one.

Algorithm A1

```
1:  Init:  the_super_frame_length = 1
2:  while (every path cannot be set up)
3:     if (a path cannot be set up at all)
4:        the_super_frame_length++
5:     for (s= N-1;s >= 1;s--)
6:        for (i=0; i <= N-1; i++)
7:           Attempt to set the path (i, s)
8:  Finally obtain the_super_frame_length
```

4.1.2. Algorithm A2

We next present the algorithm A2, which first attempts to set the path using largest elements of traffic weight matrix with respect to links, transmitters, and receivers. Here, the traffic weight matrix $C_W = \{w^{(i,s)}\}$ is defined as

$$w^{(i,s)} = \begin{cases} \frac{\sum_{k=i}^{i+s-1} n^{(k)}}{K} + \frac{s_p{}^{(i)}}{T_i} + \frac{r_p{}^{(i+s)}}{R_{i+s}}, & \text{if } c^{(i,s)} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The setup is first tried for the path with the element with a maximum value in C_W. During algorithm execution, the traffic load matrix C is updated such that the paths that having been set up are excluded. The traffic weight matrix C_W should also reflect changes of C. The algorithm is summarized in the below.

Algorithm A2

Init: `the_super_frame_length = 1`
`while (`every path cannot be set up`)`
  C_W is updated from C
  `while (`a path can be set up`)`
    Try to set the path with a maximum element of C_W
    If cannot, set the element value to be 0
  `the_super_frame_length++`
Finally obtain `the_super_frame_length`

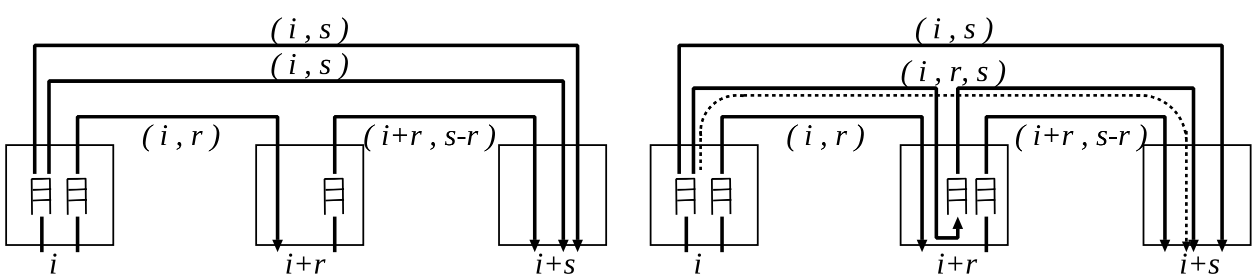
4.1.3. Algorithm A3

In the third algorithm A3, we consider the accommodation balance for the most effective use of each slot on every link. Namely, a first step of the algorithm A3 is to set the two paths between two nodes, which are located on the opposite angle of rings. Two paths between arbitrary two nodes are also chosen at the same time. When the traffic load is non–uniform, every paths are not always set up at the same time. Thus, each path should be set up independently. The algorithm is summarized in the below.

Algorithm A3

Init: `the_super_frame_length = 1`
`while (`every path cannot be set up`)`
  `if (`a path cannot be set up at all`)`
    `the_super_frame_length++`
  `for (`i`=0 ;` i `<=` $\lfloor \frac{N}{2} \rfloor - 1$`;` i`++)`
    Attempt to set each path $(i, \frac{N}{2})$ and $(i + \frac{N}{2}, \frac{N}{2})$
  `for (`s`=1;` s `<=` $\lfloor \frac{N}{2} \rfloor - 1$`;` s`++)`
    `for (`i`=0 ;` i `<=` N`-1 ;` i`++)`
      Set each path (i, s) and $(i + s, N - s)$
Finally obtain `the_super_frame_length`

4.2. Path–Splitting Access (Multi–Hop Routing)



(a) All–Optical Paths (b) Path–Splitting

Figure 5. Difference of All–Optical and Path–Splitting Accesses

In the previous subsection, we have proposed three path accommodation algorithms for all–optical access. Those aim at achieving the theoretical lower bound of the super–frame length as close as possible. However, it often fails (see the next subsection for numerical examples). In this subsection, we therefore consider the path–splitting access by dividing several paths in order to attain the lower bound more closely.

$$\begin{array}{cc} & \begin{array}{cccccccccc} 0 & 1 & 2 & \cdots & 15 & 16 & \cdots & 29 & 30 & 31 \end{array} \\ \begin{array}{c} 0\\ 1\\ 2\\ \vdots\\ 15\\ 16\\ \vdots\\ 29\\ 30\\ 31 \end{array} & \begin{pmatrix} 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \end{pmatrix} \end{array}$$

(a) C_1

$$\begin{array}{cc} & \begin{array}{cccccccccc} 0 & 1 & 2 & \cdots & 15 & 16 & \cdots & 29 & 30 & 31 \end{array} \\ \begin{array}{c} 0\\ 1\\ 2\\ \vdots\\ 15\\ 16\\ \vdots\\ 29\\ 30\\ 31 \end{array} & \begin{pmatrix} 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & \mathbf{2} \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & \mathbf{2} & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & \mathbf{2} & 1 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & \mathbf{2} & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & \mathbf{2} & 1 & \cdots & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & \mathbf{2} & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & \mathbf{2} & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \end{pmatrix} \end{array}$$

(b) C_2

$$\begin{array}{cc} & \begin{array}{cccccccccc} 0 & 1 & 2 & \cdots & 15 & 16 & \cdots & 29 & 30 & 31 \end{array} \\ \begin{array}{c} 0\\ 1\\ 2\\ \vdots\\ 15\\ 16\\ \vdots\\ 29\\ 30\\ 31 \end{array} & \begin{pmatrix} 0 & 0 & 3 & \cdots & 2 & 1 & \cdots & 3 & 3 & 1 \\ 0 & 3 & 2 & \cdots & 0 & 3 & \cdots & 0 & 1 & 2 \\ 0 & 1 & 3 & \cdots & 1 & 2 & \cdots & 2 & 2 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 2 & 0 & \cdots & 3 & 0 & \cdots & 1 & 3 & 2 \\ 0 & 1 & 3 & \cdots & 1 & 2 & \cdots & 2 & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 3 & \cdots & 1 & 2 & \cdots & 2 & 2 & 3 \\ 0 & 2 & 0 & \cdots & 3 & 0 & \cdots & 3 & 0 & 1 \\ 0 & 1 & 2 & \cdots & 0 & 2 & \cdots & 1 & 2 & 2 \end{pmatrix} \end{array}$$

(c) C_3

Figure 6. Traffic Matrices for Numerical Examples

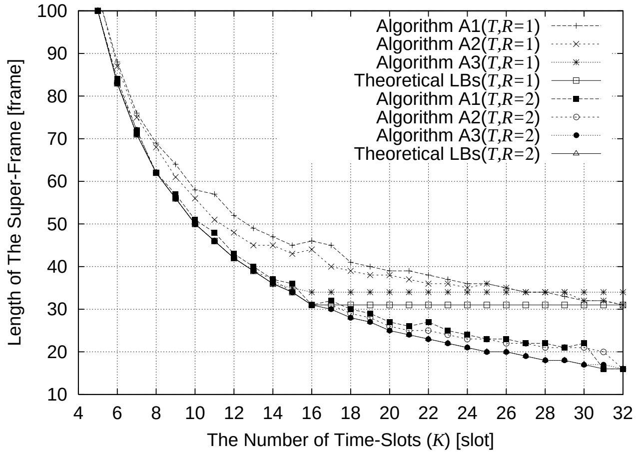
As an example, see Fig. 5. Suppose that in the case of all–optical access (Fig. 5(a)), two all–optical slots are set up between two nodes i and $i+s$ (path (i,s)), and other paths between nodes i and $i+r$ (path (i,r)), and between nodes $i+r$ and $i+s$ (path $(i+r, s-r)$) exist. If there exist more the other traffic between nodes i and $i+s$ than that between nodes i and $i+r$, and between $i+r$ and $i+s$, it is natural to split one direct optical path (i,s) into two paths as shown in Fig. 5(b). Then, we still need only three slots within the frame. We refer such a path (i,s) divided into path (i,r) and path $(i+r, s-r)$ to a splitting path (i,r,s). By splitting a path into two paths, an OE/EO conversion is necessary at node $i+r$ for the traffic between nodes i and $i+s$. It means that the packet relaying delay is incurred at node $i+r$. However, we can expect the shorter length of the super–frame if path–splitting is adequately established.

The purpose of this subsection is to propose the path–splitting algorithm, and numerical results will be presented in the next subsection. The packet delay including possible path–splitting will be provided in the next section.

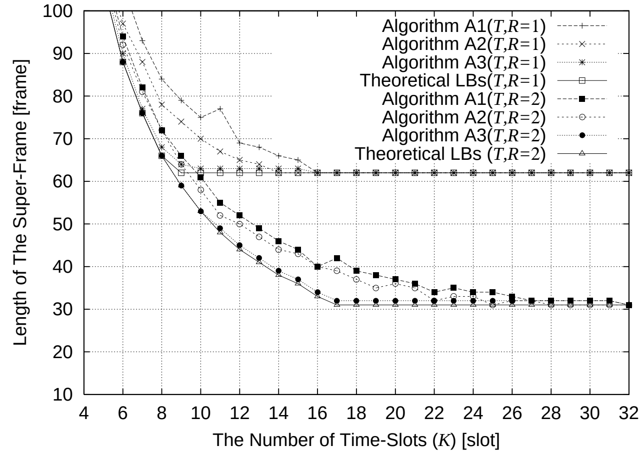
In our path–splitting method, we use the set of paths established by one of three path accommodation methods in the previous subsection. Then, we choose the path with the maximal value of C_W as a candidate of path–splitting. Then, the node with the transmitters/receivers having the lowest weight along the path is selected as its splitting point. Consequently, the splitting node $i+r$ is chosen if

$$\min_{1 \le r < s} \frac{r_p^{(i+r)}}{R_{i+r}} + \frac{s_p^{(i+r)}}{T_{i+r}}. \tag{9}$$

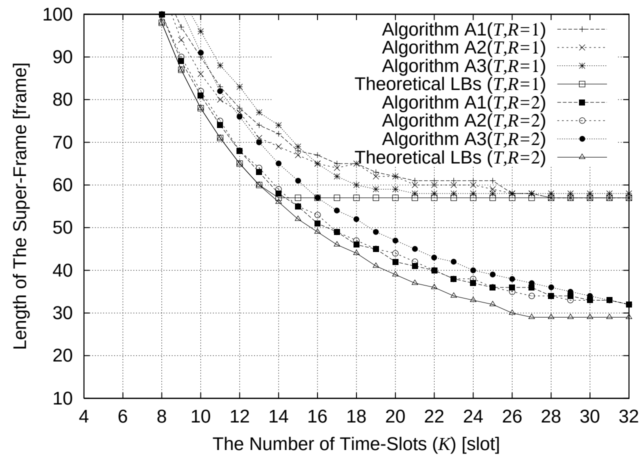
In this way, the traffic matrix is reconstructed. After our proposed algorithm (A1, A2 or A3) is applied, the path–splitting is finally performed so that the super–frame length becomes smaller than the input. The super–frame length approaches the theoretical lower bound by repeating the above procedure. If not, the iteration is terminated.



(a) The case of traffic matrix C_1



(b) The case of traffic matrix C_2



(c) The case of traffic matrix C_3

Figure 7. Comparisons of Lower Bounds and Super–frame Lengths by Three Algorithms

4.3. Numerical Examples of Algorithms and Access Methods

In this subsection, we first compare three algorithms A1, A2 and A3 presented in the previous subsection. The number of nodes, N, is fixed at 32. For the traffic load matrices, we will consider C_1 (Fig. 6(a)), C_2 (Fig. 6(b)) and C_3 (Fig. 6(c)). The characteristics of those traffic matrices are summarized below.

- C_1: a uniform traffic load.
- C_2: all paths except the ones with destination node 31 are uniform. The load of paths from any source node to destination node 31 is twice larger than that of others.
- C_3: the load of each path is randomly decided between 0 and 3.

We assume that the numbers of transmitters/receivers per node are identically set; i.e., $T = R$.

Figs. 7(a), 7(b) and 7(c) show the super–frame lengths obtained by applying three algorithms to the traffic matrix C_1, C_2 and C_3 for all–optical access, respectively. In obtaining these figures, two cases for the number of transceivers are considered; $T = R = 1$ and $T = R = 2$. The horizontal axis ($4 \leq K \leq 32$) shows the number of slots within the frame, and the vertical axis does the length of the super–frame in slots. From those figures, we can observe that the algorithm A3 gives good results for matrix C_1 and C_2. On the other hand, the algorithms A1 and A2 are better than A3 for matrix C_3 especially in the case of $T = R = 2$. That is, figures show that at least one of those algorithms can offer the good result close to the theoretical lower bounds, but we cannot find the best one always offering the shortest super–frame. It was confirmed by an extensive evaluation of the other cases with various combination of parameters. Thus, our recommendation is that all three algorithms should be performed, and the best one is chosen to set up the optical paths.

Table 1. Results by Path Splitting for Traffic Load Matrix C_2

# of $T_{ransmitters}$, $R_{eceivers}$	# of the slots (K)	Theoretical LB	All-Optical Access			Path Splitting Access					
			A1	A2	A3	A1	# of splitting	A2	# of splitting	A3	# of splitting
1	11	62	77	67	63	64	13	66	1	62	1
1	12	62	69	65	63	65	4	63	2	62	1
1	13	62	68	64	63	63	5	62	2	62	1
2	11	48	55	52	49	51	13	50	5	48	1
2	12	44	52	50	45	47	10	48	5	44	1
2	13	41	49	47	42	45	11	44	6	41	1
2	14	38	46	44	39	41	12	41	7	38	(14,1,17) 1
2	15	36	44	43	37	39	13	39	10	36	1
2	16	33	40	40	34	37	5	38	4	33	1
2	26	31	33	32	32	32	1	31	3	31	2

In the figure, it can also be observed that in several parameter regions, none of three methods can achieve the theoretical lower bounds closely. In those cases, we can expect the effect of the path–splitting access. The results against the matrix C_2 are shown in Table 1. Values in first two columns corresponds to the parameters, by which all of three path accommodation methods did not achieve good results for the super–frame length. As stated above, the algorithm A3 exhibits better results for matrix C_2 than others. It can be confirmed by the column labelled by "All–Optical Access." The last column labelled by "Path–Splitting Access" shows the results by path–splitting. We find that path–splitting access can decrease the super–frame length to the theoretical lower bound in these cases. For example, in the case of $T = R = 2$ and $K = 14$, all–optical path $(14, 17)$ is split at node 15 into two paths $(14, 1)$ and $(15, 16)$ as shown in the table. One possible problem by path–splitting is that the packet transmission delay is increased by introducing the packet relay time at the intermediate node even though the super–frame length is shortened. We therefore need to examine the packet transmission delay, which will be provided in the next section.

5. ANALYSIS OF PACKET DELAY TIMES

In this section, we analyze the packet average delay time for both all–optical and path–splitting accesses. The case of the uniform traffic load in all–optical access is first treated in Subsection 5.1. The result is then modified to derive the approximate packet delay time for the non–uniform traffic load in Subsection 5.2. The packet delay time for path–splitting access is next considered in Subsection 5.3. We will derive the packet delay time from source node i to destination node $i + s$ (i.e., on the path (i, s)). Numerical examples are finally provided in Subsection 5.4.

5.1. Packet Delay Times for the Uniform Traffic Load with All–Optical Access

By letting the capacity of the unidirectional ring be B_R [bps], one slot time denoted by t [s] is given by

$$t = \frac{(S_h + S_p) \cdot 8}{B_R}, \tag{10}$$

where S_h [byte] and S_p [byte] are the header and payload sizes of the mini–packet. The propagation delay between nodes i and $(i + s)$ is denoted by $W_p^{(i,s)}$ [s]. Further, the number of frames in the super–frame is represented by r, which has been determined by our path accommodation algorithms presented in the previous section. Then, the number of slots contained in the super–frame, D, is given by $K \cdot r$, where K is the number of time–slots per frame.

We assume that at source node i, packets destined for node $(i + s)$ arrive according to a Poisson distribution with rate $\lambda^{(i,s)}$. Hereafter, we will derive the mean packet delay time for this stream. The packet length in bytes has a general distribution with probability function f, and we represent its mean by P_B [byte]. The traffic load (in bps) for path (i,s) is then given by

$$B_f^{(i,s)} = \frac{\lambda^{(i,s)} \cdot P_B \cdot 8}{t}. \tag{11}$$

Further, we introduce the random variable P_m, representing the number of mini–packets in the packet. Its probability function, $g(n)$ $(n = 1, 2, \ldots)$, is given by

$$g(n) = \text{Prob}[P_m = n] = \sum_{x=S_p(n-1)+1}^{S_p \cdot n} f(x). \tag{12}$$

Our objective is to derive the packet delay time $W^{(i,s)}$ [s] on path (i,s), which consists of four components;

$$W^{(i,s)} = \left[\frac{D}{2} + W_q^{(i,s)} + (E[T_F] - (D-1))\right] \cdot t + W_p^{(i,s)}. \tag{13}$$

The last term, $W_p^{(i,s)}$, is the propagation delay from source node i to destination node $(i+s)$. The first term in braces is necessary because we consider the random arrival of packets, and the packet should wait the half of the super–frame in average. It corresponds to the time duration that the first mini–packet can be put on the slot assigned to that path since it reaches the head of the packet buffer. We next examine the third term in braces. The random variable T_F [slots] in the term shows the mean time to transmit all mini–packets contained in the packet from the time when the designated packet reaches the head of the queue. Since it needs the number $E[P_m]$ of super–frames, the following equation holds;

$$E[T_F] = D \cdot E[P_m]. \tag{14}$$

The subtraction of $D-1$ from $E[T_F]$ is necessary since we consider the time interval until the last mini–packet is put onto the ring. The second term of the right hand side in Eq. (13), $W_q^{(i,s)}$, corresponds to the queueing time at the source node buffer until the packet reaches the head of the queue. By applying the Pollaczek–Khinchin formula, it can be obtained by

$$W_q^{(i,s)} = \frac{\lambda^{(i,s)} E[T_F^2]}{2(1 - \lambda^{(i,s)} E[T_F])}, \tag{15}$$

where $E[T_F^2]$ is given by $D^2 E[{P_m}^2]$.

By rewriting Eq. (13) using Eqs. (14) and (15), we finally have

$$W^{(i,s)} = \left[\frac{\lambda^{(i,s)} D^2 E[{P_m}^2]}{2(1 - \lambda^{(i,s)} D E[P_m])} + D\left(E[P_m] - \frac{1}{2}\right) + 1\right] \cdot t + W_p^{(i,s)}. \tag{16}$$

5.2. Extension to the Non–Uniform Traffic Load Case in All–Optical Access

In the case of the non–uniform traffic load, two or more slots may be assigned within a single super–frame for the source/destination pair. The positions of assigned slots must depend on the path accommodation algorithm, and the intervals of slots may be irregular. Those make it impossible to derive the packet transmission time in an exact form. Hence, we introduce the assumption that assigned slots are uniformly distributed within the super–frame. More specifically, the chance to transmit the mini–packet destined for node $(i+s)$ visits source node i every $D/c^{(i,s)}$ slots. Note that D and $c^{(i,s)}$ mean the number of slots of the super–frame and the number of slots assigned to path (i,s) during the super–frame, respectively.

By the above assumption, the mean packet delay can be derived by modifying Eq. (16) as

$$W^{(i,s)} = \left[\frac{\lambda^{(i,s)} (D/c^{(i,s)})^2 E[{P_m}^2]}{2(1 - \lambda^{(i,s)} (D/c^{(i,s)}) E[P_m])} + \frac{D}{c^{(i,s)}}\left(E[P_m] - \frac{1}{2}\right) + 1\right] \cdot t + W_p^{(i,s)}. \tag{17}$$

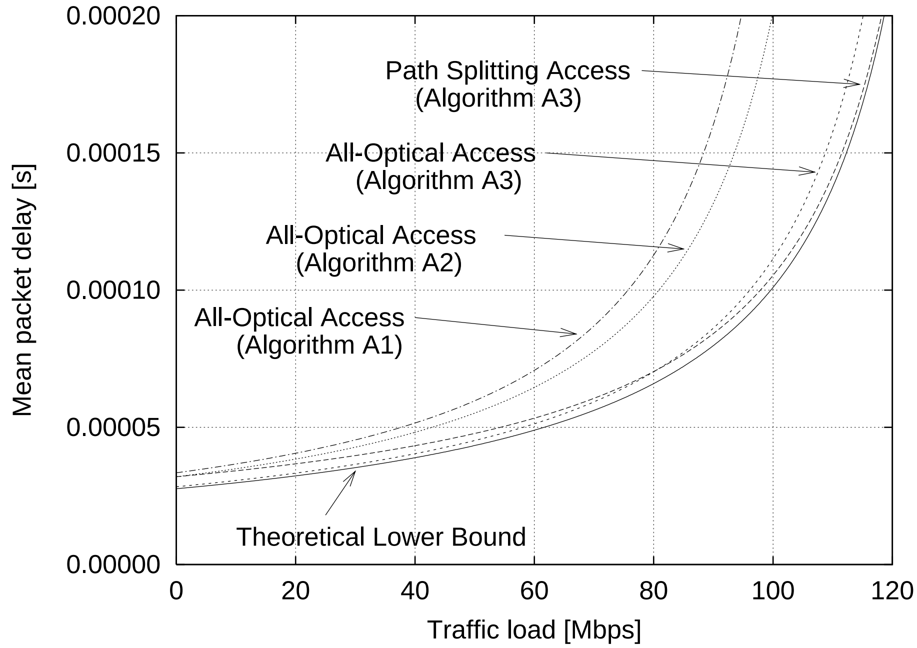


Figure 8. Comparisons of Packet Delay Times

5.3. The Case of the Path–Splitting Access

If all–optical path (i, s) is divided into two paths (i, r) and $(i + r, s - r)$, the mini–packets on the path–splitting path (i, r, s) is electronically stored at the splitting node $i + r$ after OE conversion. Each mini–packet has to wait for a next time–slot allocated for the splitting node $i + r$ for that path. We assume that those mini–packets are transmitted on each slot by distinguishing with packets by other all–optical accesses. See Fig. 5(b). Then, the packet delay time $W^{(i,s)}$ for path–splitting access is represented by

$$W^{(i,s)} = \left[\frac{\lambda^{(i,s)}(\frac{D}{c^{(i,s)}})^2 E[P_m^2]}{2(1-\lambda^{(i,s)}(\frac{D}{c^{(i,s)}})E[P_m])} + \frac{D}{c^{(i,s)}} \left(E[P_m] - \frac{1}{2} \right) + 1 + \sum_{1 \le r < s} \frac{c^{(i,r,s)}}{c^{(i,s)}} \left(\frac{D}{2c^{(i,r,s)}} + e \right) \right] \cdot t + W_p^{(i,s)}, \quad (18)$$

where $c^{(i,r,s)}$ is the number of the splitting paths (i, r, s), and e is the packet relay time at each splitting node, which is assumed to be the one super–frame length, D in the next examples.

5.4. Numerical Examples and Discussions

In the following numerical examples, we assume that the distribution of the packet size follows the geometric function;

$$f(x) = (1 - 1/P_B)^{x-1} \times 1/P_B. \quad (19)$$

The mean packet size P_B is set to be 500 [byte], and the header and payload sizes of the mini–packet, (S_h and S_p) be 2 [byte], 53 [byte], respectively. The ring capacity, B_R, is fixed at 40 [Gbps].

Figure 8 shows the results of the average packet delay time against the traffic load matrix C_2 (Fig. 6(b)). Here, the numbers of transceivers are identically set to be 2 at every node, i.e., $T = R = 2$. The number of time–slots per frame is 14 [slot] ($K = 14$). See the corresponding row of Table 1. The super–frame lengths of the theoretical lower bound, all–optical accesses by algorithm A1, A2 and A3, and path–splitting access are 38, 46, 44, 39 and 38, respectively. In the result, we set the propagation delays, $W_p^{(i,s)}$'s, to be 0 since our primary concern in this subsection is to compare all–optical accesses by three algorithms (A1, A2 and A3) and the path–splitting access.

In Fig. 8, both cases of all–optical access (by three algorithms) and path–splitting access are shown. The packet delay by the theoretical lower bound are also shown. As shown in the figure, achieving the shorter super–frame can lead to the smaller packet delay times. It can be observed by comparing the results of algorithms A1, A2 and A3. However, path–splitting should be carefully treated; when the traffic load is high, splitting the path can decrease the packet transmission time. As the traffic load becomes low, however, path–splitting increases the packet transmission delay since it introduces an extra delay at the splitting node.

6. CONCLUDING REMARKS AND FUTURE WORKS

In this paper, we have proposed and evaluated the path accommodation methods for the unidirectional OCTDM ring, and analyzed the packet transmission time. We have first derived the theoretical lower bound for the length of the super–frame, in which all paths among nodes are perfectly allocated. Three path accommodation algorithms for all–optical access are next proposed to treat the non–uniform traffic load. Path–splitting access is then treated to decrease the super–frame length. Our algorithm can achieve it, but the result should be carefully examined since the packet transmission time may be increased by introducing path–splitting. It can be checked by our analysis methods.

As future research works, the reliability issue for OCTDM rings should be investigated. We should also study on the effectiveness of the optical compression TDM/WDM where the optical compression is applied to each of multiple wavelengths in WDM.

ACKNOWLEDGMENTS

This work was supported in part by Research for the Future Program of Japan Society for the Promotion of Science under the Project “Integrated Network Architecture for Advanced Multimedia Application Systems” (JSPS-RFTF97R16301).

REFERENCES

1. K. L. Deng, K. I. Kang, I. Glesk, P. R. Prucnal, and S. Shin, “Optical packet compressor for ultra-fast packet-switched optical networks,” *Electronics Letters*, vol. 33, no. 14, pp. 1237–1239, July 1997.
2. H. Toda, F. Nakada, M. Suzuki, and A. Hasegawa, “An optical packet compressor using a fiber loop for a feasible all optical TDM network,” in *Proceedings of 25th European Conference on Optical Communication (ECOC’99)*, vol. Tu C3.7, September 1999.
3. A. Hasegawa and H. Toda, “An optical packet compressor for a feasible all optical inter-LAN TDM network,” in *Proceedings of Broadband Access and Technology, European Conference on Networks and Optical Communications (NOC’99)* (D. W. Faulkner and A. L. Harmer, eds.), pp. 233–238, IOS press, June 1999.
4. A. Hasegawa and H. Toda, “A feasible all optical soliton based inter-LAN link using time division multiplexing,” *IEICE Transactions on Communications*, vol. E81-B, pp. 1681–1686, August 1998.
5. N. S. Patel, K. L. Hall, and K. A. Rauschenbach, “Optical rate conversion for high-speed TDM networks,” *IEEE Photonics Technology Letters*, vol. 9, no. 9, pp. 1277–1279, September 1997.
6. B. Y. Yu, P. Toliver, R. J. Runser, K. L. Deng, D. Zhou, I. Glesk, and P. R. Prucnal, “Packet-switched optical networks,” *IEEE Micro*, vol. 18, no. 1, pp. 28–38, January-February 1998.
7. K. Gokyu, K. Baba, and M. Murata, “On path accommodation methods for optical compression TDM ring,” in *Proceedings of Workshop on Optical Networks 2000*, January 2000.
8. O. Gerstel, P. Lin, and G. Sasaki, “Wavelength assignment in a WDM ring to minimize cost of embedded sonet rings,” in *Proceedings of the Conference on Computer Communications (INFOCOM’98)*, pp. 94–101, March 1998.
9. X. Zhang and C. Qiao, “On optimal scheduling and cost effective design in WDM rings,” *IEEE/LEOS Broadband Optical Networks*, Paper TuB3, Aug. 1996.
10. G. Li and R. Simha, “Efficient routing to reduce the cost of add-drop multiplexers in WDM optical ring networks,” *Proceeding of the SPIE Conference on All-Optical Networking 1999: Architecture, Control, and Management Issues*, vol. 3843, pp. 258–267, Sept. 1999.
11. X. Zhang and C. Qiao, “On scheduling all-to-all connections and cost-effective designs in WDM rings,” *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 435–445, June 1999.
12. E. Coffman, M. R. Garey, and D. S. Johnson, “An application of binpacking to multiprocessor scheduling,” *SIAM J. Computing*, vol. 7, pp. 1–17, Feb. 1978.
13. M. R. Garey, R. L. Graham, and D. S. Johnson, “Performance guarantees for scheduling algorithms,” *Oper. Res.*, vol. 26, pp. 3–21, Jan. 1978.

14. M. S. Borella and B. Mukherjee, “Efficient scheduling of nonuniform packet traffic in a WDM/TDM local lightwave network with arbitrary transceiver tuning latencies,” *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 923–934, June 1996.