PAPER

Maintaining Packet Order in Reservation-Based Shared-Memory Optical Packet Switch

Xiaoliang WANG[†a)], **Xiaohong JIANG**[†b)], *Nonmembers*, *and* **Susumu HORIGUCHI**[†c)], *Member*

SUMMARY Shared-Memory Optical Packet (SMOP) switch architecture is very promising for significantly reducing the amount of required optical memory, which is typically constructed from fiber delay lines (FDLs). The current reservation-based scheduling algorithms for SMOP switches can effectively utilize the FDLs and achieve a low packet loss rate by simply reserving the departure time for each arrival packet. It is notable, however, that such a simple scheduling scheme may introduce a significant packet out of order problem. In this paper, we first identify the two main sources of packet out of order problem in the current reservation-based SMOP switches. We then show that by introducing a "last-timestamp" variable and modifying the corresponding FDLs arrangement as well as the scheduling process in the current reservation-based SMOP switches, it is possible to keep packets in-sequence while still maintaining a similar delay and packet loss performance as the previous design. Finally, we further extend our work to support the variable-length burst switching.

key words: *OPS, Shared-Memory Optical Packet switch, Mis-sequence*

1. Introduction

The explosive increase of Internet traffic requires fast and high capacity switching networks. All-optical switching, where the data transmission is in the optical domain but the switching control can be in the electrical domain, is able to meet these requests since it eliminates the quite expensive optical-electronic-optical conversion and provides us an opportunity to make good use of the enormous bandwidth of optical networks [1], [2]. Time sliced (synchronous) optical switching without wavelength conversion is a simple and cost-effective technology for implementing all-optical packet switching [3]–[6], where contending packets are temporarily buffered and forwarded at a later time slot. In optical networks, fiber delay lines (FDLs) are usually adopted to delay packet since the optical-RAM buffer is not available yet. Unlike the traditional electronic memories in which packets can be randomly read and written, a packet entering a fiber delay line must emerge a fixed time later and can not be removed before that time. As such, the implementation of large buffers requires a large number of fiber delay lines and thus a high hardware cost. To reduce the amount of required memory in a packet switch while guaranteeing a desired level of throughput or packet loss rate, numerous shared-memory optical packet switches have been proposed in the literature, see, for example, [2], [4], [5], [7], and the references therein.

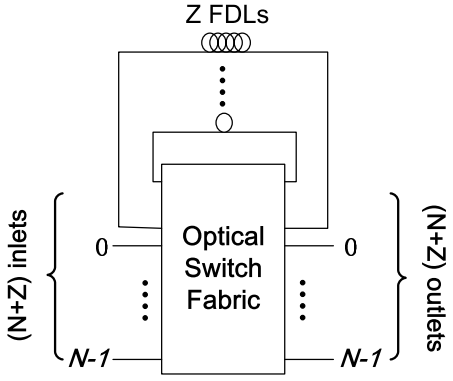

Fig. 1 Shared-Memory Optical Packet (SMOP) switch architecture.

A typical Shared-Memory Optical Packet (SMOP) switch architecture is illustrated in Fig. 1, which was first proposed by Karol in [4] to support fixed length optical packet switching (ATM switching). But for Internet traffic, this structure is also suitable. Similar to most of the current electronic routers, if the packets have variable sizes, they are segmented into fixed size blocks during arrival and reassembled at departure. The Fig. 1 shows clearly that to construct a single stage $N \times N$ SMOP switch with Z shared feedback FDLs, we actually need a $(N+Z) \times (N+Z)$ space switch, where the outputs (inputs) of FDLs and switch are collectively called the inlets (outlets) of the switch. Under the feedback FDLs design, a packet will be either directly transported to its output port or forwarded to one of the Z fiber delay lines.

For contention resolution, a scheduling algorithm is usually required to direct packets through the switch. In [4], two scheduling algorithms (Non-FIFO/FIFO) for the SMOP switch have been proposed. In the Non-FIFO algorithm, the controller first schedules the recirculation packets (from the longest delay line to the shortest delay line) and then schedules the newly arrived packets. If a packet failed in the contention and can not be routed to its output, the failed packet will be sent to the delay line where there is the fewest number of packets destined for the same output at the emerge time slot. If there are multiple delay lines which satisfy the above condition, the controller will choose the shortest delay line. By doing so, it helps to "balance" the output addresses of packets at each time slot. In the FIFO algorithm, a priority is assigned to each arrived packet based on its arrival

Manuscript received January 11, 2008.
Manuscript revised April 25, 2008.

[†]The authors are with the Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980-8579 Japan.
a) E-mail: waxili@ecei.tohoku.ac.jp
b) E-mail: jiang@ecei.tohoku.ac.jp
c) E-mail: susumu@ecei.tohoku.ac.jp

DOI: 10.1093/ietcom/e91–b.9.2889

order. In addition, a FIFO list for all packets in each flow[†] is maintained to indicate the position of buffered packets in FDLs. Based on the FIFO list, this algorithm ensures that a packet can be scheduled only if all other packets with higher priorities have left the switch. Since the above two algorithms can not guarantee the departure time for packets that are lost in the contention, the number of packet recirculation is unpredictable in advance. The simulation results in [4] indicate that the maximum number of recirculation in Karol's algorithm can be as high as 10 times [4], which is undesired since the optical signal will be significantly attenuated with such number of recirculations.

To alleviate the above multi-circulation problem, S.Y. Liew et al. proposed reservation based scheduling algorithms for the single-stage shared-FDL switch in [5]. The reservation scheme performs not only the output port matching for current time slot but also the FDLs assignment for the entire journey of a delayed packet so that it can be scheduled to match with the desired output port in the future time slots. The number of packet recirculation is constrained by the maximum number of FDLs delay operation, k. If the delay path to the right destination is unavailable, the packet will be dropped to avoid any resources occupation.

It is notable, however, that a significant problem with the reservation based algorithm is that the packets may be mis-sequence (to be explained in Sect. 2). In the traditional electronic domain, the mis-sequence problem can be easily solved through introducing additional re-sequence buffer at output ports. Unfortunately, the lack of optical random access memory is one of the problems in optical switching. Designing a re-sequence buffer using switch and FDLs is costly and introduces extra delay operation [8]. In this paper, we focus on developing a framework to prevent packet from mis-sequence while maintaining a similar packet loss rate and delay performance as the original reservation-based SMOP switches.

The rest of the paper is organized as follows: Sect. 2 provides a review of the reservation scheduling algorithms and identifies the sources of packet mis-sequence problem. Sect. 3 first defines the "last-timestamp" variables to prevent from packets mis-sequence and then introduces our framework to ensure the packet loss rate and delay performance by modifying the FDLs lengths arrangement as well as the the scheduling process in the current reservation-based algorithms. Section 4 presents some numerical results by simulation and compares the complexity of different schemes. Section 5 extend our work to support variable-length burst switching. Finally, Sect. 6 concludes the paper.

2. The Reservation Scheme and Packet Mis-Sequence Problem

In this section, we first explain the reservation scheduling algorithm and then show that the packet mis-sequence problem is actually caused by two reasons, namely the "restriction of FDLs" and "restriction of algorithm". The notations employed in this paper are listed in Table 1.

2.1 Reservation Scheme

Three reservation algorithms were proposed in [5]: sequential FDL assignment algorithm (SEFA), multi-packet FDL assignment algorithm (MUFA) and parallel iterative FDL assignment algorithm (PIFA). The SEFA algorithm considers the FDL assignment for only a single packet. The MUFA algorithm extends the SEFA algorithm to process multi-packets simultaneously. For the PIFA algorithm, it is a distributed algorithm which uses plenty of "request-grant-filter-accept" control information among nodes in the slot transition diagram. In this paper, we take the MUFA algorithm as an example to introduce the reservation scheme, because this algorithm is simple and able to process multi packets simultaneously.

In the MUFA algorithm, the maximum packet delay is constrained by parameter F. Fiber delay lines are allowed to have the same delay values where the delay values are distributed among $2^0\ T_{cell}$, $2^1\ T_{cell}$,..., $2^{f-1}\ T_{cell}$ ($f = \log_2 F$), as illustrated in Fig. 2(a). For example, if $F = 128$ and $N = Z = 32$, there are 5,5,5,5,4,4, and 4 FDLs with delay values of 1,2,4,8,16,32 and 64 time slots, respectively. To forward packets in a SMOP switch, the MUFA algorithm maintains a configuration table T for marking all the possible FDL routes and indicating the decision of scheduling. As shown in Fig. 3, the table T adopts $(N + Z)$ rows and F columns to represent the $(N+Z)$ outlets occupation in the future F time slots. Initially, all the blanks are set to 0, which indicates all the output ports and FDLs are free. The procedure of MUFA algorithm can be explained by using a slot

Table 1 Notations used in this paper.

Variables	Comments
N	number of input/output ports
Z	number of feedback FDLs
F	maximum delay value, $F-1$ time slots
K	maximum number of packet circulation
D_a	length of FDL a
T_{cell}	duration of a time slot
$T_k(t)$	the $level-k$ node with $t\ T_{cell}$ delay in G
$flow(i, j)$	the packet flow from input i to output j
$P^m_{i,j}$	the m^{th} arrived packet of flow (i, j)
$T_{last}(i, j)$	the output time slot subscribed by the last packet of (i, j)

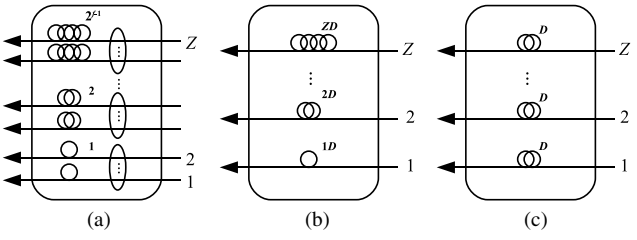

Fig. 2 Fiber delay lines configuration.

[†]The $flow(i, j)$ consists of packets that have the same source i and the same destination j.

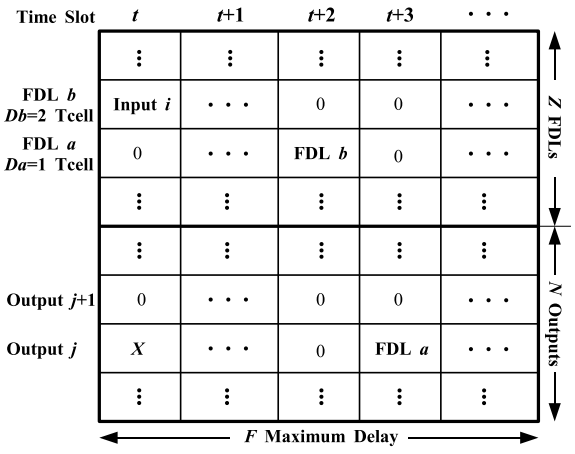

Fig. 3 The configuration table T.

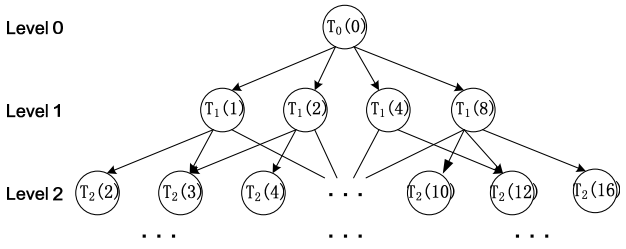

Fig. 4 The slot transition diagram G.

transition diagram G. As shown in Fig. 4, if FDL a is idle at time slot t then there is an arc from $T(t)$ to $T(t + D_a)$ in the transition diagram.

At each time slot, the MUFA algorithm uses breadth-first search-based algorithm to find routes from G. The rule is that the parent node will make decision for its accessible child nodes, i.e. the node $T_{k-1}(\tau)$ makes decision for nodes $T_k(t)$ when the route from $T_0(0)$ to $T_k(t)$ via node $T_{k-1}(\tau)$ is available. If there is a request whose destination port is idle at time slot t, the route from $T_0(0)$ to $T_k(t)$ will be marked in configuration table T for this request. For the example considered in Fig. 3 (where symbol X means this channel is not idle), we can see the scheduling result of packet $P^m_{i,j}$ that arrived at time slot t. The delay path of this packet is as follows: First, this packet is going to be buffered in FDL b with length 2 T_{cell}. Then, at time slot $t + 2$, the input port of FDL a will be scheduled to connect to the output port of FDL b, and this packet will enter FDL a with delay 1 T_{cell}. Finally, at time slot $t + 3$, it is possible for the output port j to "read" packet $P^m_{i,j}$ from FDL a. This process can also be logically represented by slot transition diagram G in Fig. 4 as $T_0(0) \rightarrow T_1(2) \rightarrow T_2(3)$.

2.2 Packet Mis-Sequence Problem

It is notable that the reservation algorithm does not consider the packet mis-sequence problem, since the packets departure order may be reversed due to the contention of finite FDLs. Take the configuration table in Fig. 5 as an example,

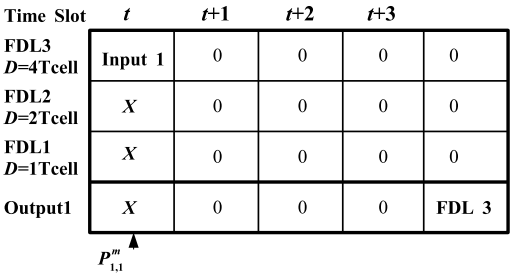

Fig. 5 Mis-sequence in a SMOP switch.

the number of delay lines is $Z = 3$ and the lengths are 1, 2 and 4 T_{cell} respectively based on the exponential distribution. Assume that a packet $P^m_{1,1}$ arrived at time slot t, but both the output port 1 and the FDL with length 1 and 2 T_{cell} were not available (due to the subscription by packets arrived at previous or current time slots). Thus, the only choice is the idle FDL with a larger delay value 4 T_{cell}. After the arbitration, we can find a gap appears from time slot $t + 1$ to $t + 3$ in the row of output 1. If a packet that belongs to the same flow happened to arrive at this time interval and filled the gap, the mis-sequence happened in $flow(1, 1)$. Because this kind of mis-sequence is caused by the limitation of FDLs, we use *the restriction of FDLs* to refer to such cause.

Another cause of the packets mis-sequence problem is *the restriction of algorithm*. Based on the MUFA algorithm, the parent node is going to make decision for its child nodes. It is possible that a larger delay route is selected rather than a smaller one, even they have the same number of delay operations, because the parent node of the former route has a smaller index [5]. As shown in Fig. 4, the delay route $T_0(0) \rightarrow T_1(1) \rightarrow T_2(5)$ may be selected prior to the route $T_0(0) \rightarrow T_1(2) \rightarrow T_2(3)$ since the node $T_1(1)$ in $level - 1$ is read before node $T_1(2)$. In this case, it is also possible to bring a gap in the configuration table T.

3. Maintaining Packet Order in SMOP Switch

The analysis in Sect. 2 indicates that the mis-sequence problem may happen in a SMOP switch due to the existence of gap in configure table T. It is easy to see that a large gap will increase the probability of packets mis-sequence and deteriorate the performance of packet loss rate. To avoid packets mis-sequence in reservation based scheme, we define the *last-timestamp* variable $T_{last}(i, j)$ to remember the furthest output time slot subscribed by the last arrived packets of flow(i, j). When making decision for unfulfilled requests in node $T_k(t)$ of G, we only need to care about those requests which satisfy $T_{last}(i, j) < t$ and output j is idle at time slot t. We call these requests *entitled requests* in the rest of this paper. By doing so, the assigned departure time slots for entitled requests in each flow will not violate the FIFO constraint. However, this method may aggravate the performance in terms of packet loss rate and delay, because the gap in T may be heavily extended. In the following, we will illustrate our approaches to reduce the gap in configu-

ration table.

3.1 Alleviating the "Restriction of FDLs"

Notice that under the exponential distribution of FDLs lengths, the MUFA algorithm will introduce a large gap in T and add the delay operations in switch. Consider the setting given in [5], the lengths of delay lines are distributed among $2^0\ T_{cell}$, $2^1\ T_{cell}$,..., $2^{f-1}\ T_{cell}$ ($f = \log_2 F$), as illustrated in Fig. 2(a). Suppose a packet $P^m_{i,j}$ arrived at time slot t and $T_{last}(i, j) = t + 8$, if the fiber delay lines with length $8\ T_{cell}$ are not idle, the next choice is the node $T_1(16)$ in *level*-1 of G, which means the packet $P^m_{i,j}$ has to be delayed 16 time slots via the delay line with length $16\ T_{cell}$. This implies a high probability of packet mis-sequence. In addition, since the only way to obtain a path of odd time slot delay is to combine the delay line of length $1\ T_{cell}$ with other FDLs, the number of delay operations may be increased. Therefore, it is more reasonable to return to the linear distribution of FDLs lengths, $D, 2D, 3D, 4D, \cdots$ where $D \geq 1$, see Fig. 2(b) which can decrease the probability of introducing a large gap in T. In the next section, the packet loss rate is compared among different D in the MUFA algorithm but adopting $T_{last}(i, j)$ to maintain packets order. We can see the lowest packet loss rate is achieved when $D = 1$. This result suggest that the selection of linear distribution of FDLs lengths, $1, 2, 3, 4, \cdots$, is a good choice in the SMOP switches.

In the slot transition diagram, for each node, the number of child nodes equals to the number of the lengths of FDLs. Let $P(i)$ be the number of FDLs with length i. There are $f = \log_2 F$ child nodes for each node in the case of exponential distribution but there are Z ($Z = \sum P(f)$ for all f) child nodes for each node in the case of linear distribution. Thus, the size of the slot transition diagram becomes lager. Since the complexity of algorithm is related to the size of slot transition diagram, by using the linear distribution of FDLs length, the time complexity is increased in the MUFA algorithm which uses $last - timestamp$ to constrain packets FIFO (called FIFO MUFA algorithm). To overcome this problem, based on the observation of simulation results in Fig. 7 that the distribution of the number of delay operations involves in 2, we will constrain the maximum number of packet circulation equals to 2 (i.e. $K = 2$).

On the other side, there is another kind of setting of fiber delay lines named F-FDL [1] where all the FDLs are assigned with fixed delay D (≥ 1), see Fig. 2(c). This setting is not suitable for the SMOP switch because a large amount of recirculations is required for contention resolution. Note that it is not advisable to solve the restriction of FDLs by adding too many FDLs, because it may heavily increases the hardware cost. Hence this paper focuses on the case that the number of FDLs equals to the number of input/output ports.

Table 2 Notations used for SMUFA algorithm.

Variables	Comments
SET_d	"The set of destination nodes" contains those nodes $T_k(t)$ that packet is possible to be sent to the output port at time slot t. Initially all the notes belong to "the set of destination nodes."
SET_t	"The set of transfer nodes" contains those nodes that failed in the output ports matching and can only be used as the 'parent nodes' to help finding longer delays, such as those nodes $T_k(t)$ who have the same delay value but unmatched at higher levels.
$N(k)$	Nodes in level k.
$route[i]$	$route[i] = 1$ means the route from $T_0(0)$ to $T_k(i)$ is available otherwise $route[i] = 0$.
$match[i]$	$match[i] = 1$ means a newly arrived packet successfully find the route to its output without violating the FIFO constrain at node $T_k(i)$ otherwise $match[i] = 0$.

3.2 Eliminating the "Restriction of Scheduling Algorithm"

First of all, we give the principles for the packet scheduling: 1) minimum delay operations; 2) select the delay paths in ascending order in each level of G so as to decrease the packet delay and the gap in T. The first reason why we shall guarantee the minimum delay operations is that optical signal gets attenuated when it is switched [2]. Another reason is that multi delay operations will subscribe the FDLs resources in future time slots, which increases the probability of contention of future time slots and results in the increment of packet loss rate.

Based on the above principles and linear distribution of FDLs length, we propose the following Sequence MUFA algorithm (SMUFA). In SMUFA, each node in G is required to reserve a list of all the effective routes from $T_0(0)$ and the nodes in each level will be read twice. When accessing a node $T_k(t)$, the algorithm first checks whether those routes from $T_0(0)$ to $T_k(t)$ are available, then matches the unfulfilled requests according to the $T_{last}(i, j)$ at time slot t. After the matching process of all the nodes in level k, those unmatched nodes will copy their effective routes to their child nodes in level $k + 1$ if the corresponding FDLs is available, and then finding the available route from nodes in level $k + 1$ for those unfulfilled requests. This searching procedure may be terminated early if all the requests have found their routing paths. By doing so, the FDL route is directly decided by the node itself but not from the parent node and eliminating the "Restriction of Scheduling Algorithm."

The notations employed for explaining the procedure of SMUFA algorithm are shown in Table 2.

Search Procedure of SMUFA

$k := 0$;
assign direct connections from input ports to output ports for new arrived entitled requests;
update configuration table T;
for $k := 1$ to K **loop**

```
    i := k;
    while (node T_k(i) ∈ N(k))
        if (node T_k(i) ∈ SET_d) then
            if (route[i] = 1 and match[i] = 1) then
                update configuration table T;
                i := i + 1; continue;
            else
                put node i to SET_t;
            end;
        end;
    i := i + 1;
    end;
    i := k;
    while (unmatched node T_k(i) ∈ N(k))
        copy effective routes to child nodes in accordance
        with the available FDLs;
        i := i + 1;
    end;
end;
```

4. Computer Simulation and Numerical Results

In this section, we first show some properties and evaluate the performance of SMOP switch through simulation. Then the complexity of different schemes are compared. In simulation, the packets arrive according to a Bernoulli arrival process and are uniformly distributed among the output ports. The traffic load is the ratio of the injection traffic rate to the capacity of port. The switch size N and number of delay lines Z equals to 32 in our simulation model, which are the same as that in [5].

4.1 Performance Evaluation

We first check the distance of packets out-of-order in MUFA algorithm. Setting the maximum delay value $F = 128$, the simulation result shows that the maximum distance of mis-sequence is 10 time slots under uniform traffic. When applying the same setting of the network but using a special unbalance traffic [9], which means 50% packets from input i ($i \leq N$) are send to output i and the other 50% packets are uniformly distributed to all of the output ports, the maximum distance of mis-sequence is 55 time slots. It means the packets mis-sequence introduced by MUFA algorithm can be a significant problem in the switching network.

In Fig. 6, the performance of packet loss rate is compared among MUFA algorithms with FIFO constrain but under different distributions of FDLs length. The curves show that the packet loss rate is the lowest when $D = 1$ and increases with the increment of D. The reason is similar to that of the exponential distribution case. With the increment of D, the probability of introducing a large gap in configuration table T increases as well. To guarantee the sequence of packets, these gaps can never be used by packets belong to the same flow, which implies an increment of packet loss rate. Thus, to achieve a better performance of packet loss rate, the $D = 1$ linear distribution of FDLs length is adopted in our SMOP switch.

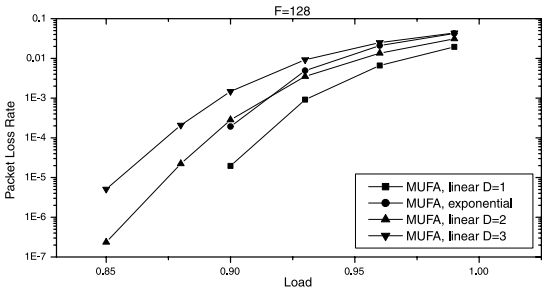

Fig. 6 Comparison of packet loss rate of MUFA with different FDL length distribution.

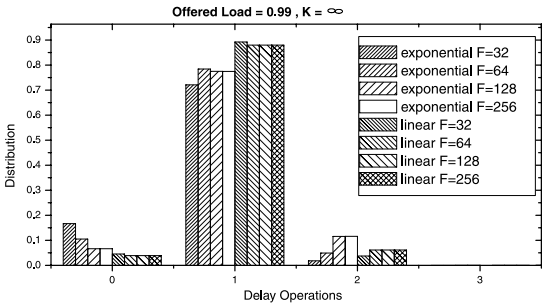

Fig. 7 Distribution of the number of delay operations in FIFO MUFA algorithm.

Figure 7 compares the number of delay operations by using the MUFA algorithm with FIFO constrain under both the exponential distribution and the linear distribution of FDLs lengths. We can see that most packets involve fewer than 3 delay operations with different maximum delay values, F. Since the maximum distance of mis-sequence is 10 packets through our simulation, by using the exponential distribution of delay lines, $1, 2, \ldots, 32, 64$, usually it is possible for the arrived packets to find a route within 1 circulation for each flow. Besides that, since the void slots (gap) can be used by packets belong to other flows, it also decreases the requirement of multi-circulation. When applying the linear distribution of FDLs, the small granularity gives the packets more choice to avoid mis-sequence, the number of 1 delay operation slightly increases. Figure 8 shows the number of delay operations by using the new SMUFA algorithm, where the distribution of delay line is linear distribution. The required number of delay operations is less and equal to 3 and most of the contention can be solved within 1 delay operation. It is worth noting that, when $F > 128$, the number of assigned delay operations are almost the same. The reason is that the breadth-first search-based algorithm search nodes from left to right in each level of the slot transition diagram G. The FDL routes always combine short delay lines with long delay lines, which reduce the probability of achieving a large delay combined by long delay lines in future time slots.

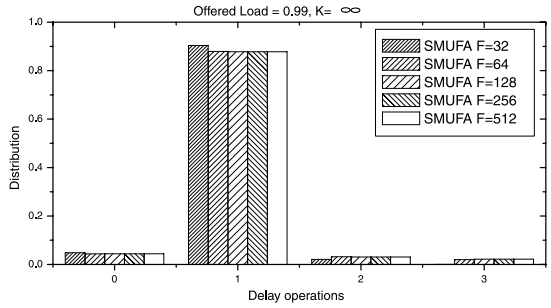

Fig. 8 Distribution of the number of delay operations in SMUFA algorithm.

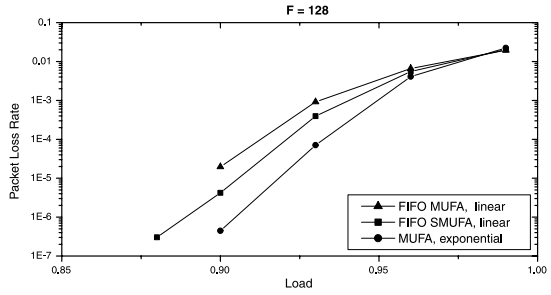

Fig. 9 Comparison of packet loss rate with different algorithms.

Figure 9 shows the packet loss rate under the processes of our solution. When using the linear distribution to replace the exponential distribution of the lengths of fiber delay lines and constraining packets first-in-first-out in each flow, we can see that the packet loss rate of MUFA algorithm is close to 10^{-5} at a 0.9 traffic loading (marked as *FIFO MUFA linear*). Further, the SMUFA algorithm which guarantees the nodes are orderly read in each level of G achieves nearly 10^{-7} packet loss rate at the load of 0.88 when $K = 3$ (marked as *FIFO SMUFA linear*). The average packet delay in Fig. 10 indicates that the MUFA and SMUFA algorithm can both achieve a low packet delay. Because of the FIFO property, the SMUFA algorithm needs a little higher delay than the MUFA algorithm under light traffic. Overall, the SMUFA algorithm achieves a similar delay and packet loss rate as MUFA algorithm but keeps packets of a flow in sequence.

Figure 11 shows the packet loss rate under different FIFO algorithms. Usually, to guarantee the packets in-sequence, the idea is that a packet can never be sent out until the packets arrived early have already been scheduled [1], [2]. If we only introduce the *last-timestamp* variable to avoid packets mis-sequence in the original MUFA algorithm (marked as *FIFO MUFA exponential*), the performance heavily deteriorate. Compared with SMUFA algorithm, the Karol FIFO algorithm achieves a better performance under higher traffic load but requires at most 10 delay operations.

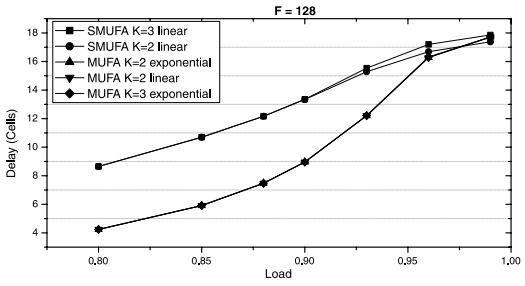

Fig. 10 Comparison of packet delay.

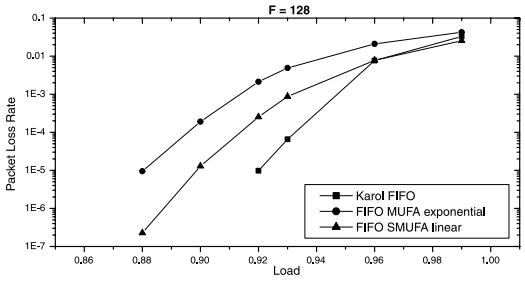

Fig. 11 Comparison of FIFO algorithms in SMOP switch.

Table 3 A complexity comparison among algorithms.

Scheduling algorithm	Karol	MUFA	SMUFA
Time complexity	$O(Z^2)$	$O((\log F)^2)$	$O(Z^2)$

4.2 Complexity Comparison

In the SMUFA algorithm, each node in slot transition diagram will be read twice. In the first round, the controller matches the unfulfilled packets and in the second round the controller checks the remain effective routes and copy the list of routes to all of its child nodes. According to the linear distribution of FDLs, there are Z nodes in each level of G, hence the complexity of SMUFA algorithm is $O(Z^2)$.

Then, we compare the time complexity of Karol algorithms [4], the MUFA algorithm and the SMUFA algorithm in the SMOP switch. In Karol algorithm, which also apply the linear distribution of FDLs, both of the non-FIFO and FIFO schemes require to find the shortest delay line that has the fewest packets destined for the same output port for the newly arrived packet or unscheduled recirculated packet. Thus the complexity is $O(Z^2)$, where Z is the number of delay lines. For the MUFA algorithm, as mentioned in [5], the complexity is related to the number of nodes in slot translation diagram. According to the exponential distribution of FDLs, the complexity is $O((\log F)^2)$, where F is the maximum delay value. Therefore, we have the results in Table 3.

Here, we also compare the implementation complexity of re-sequence buffer with our SMUFA algorithm (see Table 4). One possible way to implement the optical re-sequence buffer (Sorter) is to apply the optical fiber delay line based Time Slot Interchanger (TSI) [8], [10], [11], where the TSI is an one-input one-output network element

Table 4 A comparison between optical sorter and SMUFA.

	Sorter	SMUFA
Hardware cost	$2 \times (2 \log B - 1) \times N$	–
Time complexity	–	$O(Z^2)$
Delay operation	$2 \times (2 \log B - 1)$	–

that is capable of interchanging the positions of B arriving time slots according to any permutation. The practical complexity of the TSI construction is $O(\log B)$ (in fact $2 \log B - 1$ 2×2 switches in [10]). Assume the maximum distance of packets out-of-order in the original MUFA algorithm is B (such as 10 time slots under uniform traffic). The implementation of Sorter in [8] requires two-stage TSI to make it non-blocking and work conserving. Thus the complexity of Sorter for N output ports is $2 \times (2 \log B - 1) \times N$. Besides that, since the TSI is constructed by a concatenation of $2 \log B - 1$ 2×2 switches, the Sorter will add extra $2 \times (2 \log B - 1)$ delay operations. It is undesirable in the SMOP switching network. On the other hand, the SMUFA algorithm changes the control scheme but did not add any hardware cost and restricts the number of delay operation, which satisfies the principle of original MUFA algorithm.

5. Variable Length Burst Scheduling

In the field of telecommunication and computer communication, data traffic requires network to accommodate variable-length bursts which contain an integer number of fixed size packets. The burst length information is obtained by reading the burst headers when they first arrive to the switching network. In SMOP switch, a buffered burst with length L will occupy an optical delay line in continuous L time slots. Since the non-reservation scheduling algorithm in [4] can not guarantee the departure time for the burst that was lost in the contention, the multi circulations of variable bursts will occupy plenty of FDLs and may significantly increase the packet loss probabilities in the SMOP switches. On the other hand, the reservation based algorithm provides us an approach to finding a delay route for the whole burst which avoids the waste of FDLs. In this section, we will extend our sequence scheme to support the variable length burst switch.

The main idea for supporting variable length burst in SMUFA algorithm is to revise the matching procedure in the nodes of G. Suppose a burst with length L that belongs to $flow(i, j)$ arrives at time slot 0, if an available route $T_0(0) \rightarrow T_1(\tau) \rightarrow T_2(t)$ is found using the original SMUFA algorithm, we need to check if the vacancies in T are large enough to afford consecutive L T_{cell} in row of FDL τ behind time slot 0, in row of FDL $(t - \tau)$ behind time slot τ and in row of output j behind time slot t. If all of these vacancies are bigger than L T_{cell}, the burst can be accepted, otherwise try other routes. If such kinds of routes are not exist, the whole burst will be dropped and does not occupy any resources.

To verify the performance for supporting variable length burst, we further study packet loss rate by simulation. In this simulation, we consider the variable-length bursts generated by an ON-OFF model that the duration of the ON period is specified by the burst size distribution and the duration of the OFF period depends on the traffic load. The burst size is chosen according to the typical Ethernet traffic as 1 (resp. 16 and 38) T_{cell} with probability 0.35 (resp. 0.45 and 0.2) [9]. The distribution of burst destination is uniform and $N = Z = 64$. Figure 12 shows the packet loss rate for the variable length burst. When $K = 2$ and $F = 128$, the packet loss rate increases from 3×10^{-3} to 0.2 with the increment of load from 0.8 to 0.99. If we allow $K = 3$ and $F = 256$, the packet loss rate can achieve 10^{-4} under the traffic load of 0.8. We can see that the performance is poor compared with the fixed packets-based switching under the same configuration of FDLs. It is because the continuity constraint of variable length bursts requires a number of continuous time slots in fiber delay lines which causes far more frequent contention than that of packets with small granularity even though the buffer is far from full.

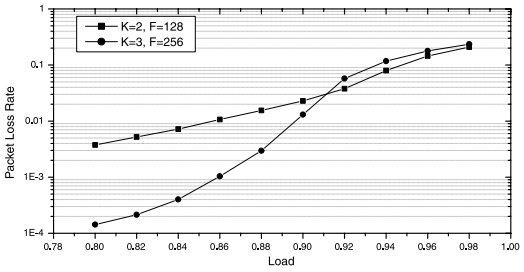

Fig. 12 Packet loss rate of SMUFA supporting variable length burst.

6. Conclusion

In this paper, we identify the two reasons that may cause and aggravate the packets mis-sequence problem in the reservation based algorithm of SMOP switch: the restriction of FDLs and the restriction of algorithm. Based on the analysis, we first defined the *last-timestamp* variable to avoid packets out-of-order, then modified the FDL length distribution and proposed an improved algorithm SMUFA to guarantee packet loss and delay performance. Through simulation, we demonstrate that our approach achieves an similar packet loss rate and delay performance as the original reservation algorithm but keeps packets of a flow in-sequence. Finally, we extended our work to support variable-length burst switches.

References

[1] F. Callegati, W. Cerroni, G. Corazza, C. Develder, M. Pickaver, and P. Demeester, "Scheduling algorithms for a slotted packet switch with either fixed or variable length packets," J. Photonic Network Communications, vol.8, no.2, pp.163–176, Sept. 2004.
[2] T. Zhang, K.J. Lu, and J.P. Jue, "Shared buffering in optical packet-switched networks," IEEE J. Sel. Areas Commun., vol.24, no.4, pp.118–127, April 2006.
[3] C.C. Chou, C.S. Chang, D.S. Lee, and J. Cheng, "A necessary and

sufficient condition for the construction of 2-to-1 optical fifo multiplexers by a single crossbar switch and fiber delay lines," IEEE Trans. Inf. Theory, vol.52, no.10, pp.4519–4531, 2006.

[4] M.J. Karol, "Shared-memory optical packet (ATM) switch," SPIE: Multigigabit Fiber Communications Systems, vol.2024, pp.212–222, July 1993.

[5] S.Y. Liew, G. Hu, and H.J. Chao, "Scheduling algorithms for shared-fiber-delay-line optical packet switches, part i: The single-stage case," J. Lightwave Technol., vol.23, pp.1586–1600, April 2005.

[6] J. Ramamirtham and J. Turner, "Time sliced optical burst switching," Proc. 22nd IEEE INFOCOM 2003, vol.3, pp.2030–2038, 2003.

[7] S. Yao, B. Mukherjee, and S. Dixit, "Advances in photonic packet switching: An overview," IEEE Commun. Mag., vol.38, no.2, pp.84–94, Feb. 2000.

[8] P.J. Lin and A. Narula-Tam, "Cell-sorting device for creating synchronous variable-length optical packet switches," J. Lightwave Technol., vol.21, no.4, pp.893–903, April 2003.

[9] Y.T. Chen, C.S. Chang, J. Cheng, and D.S. Lee, "Feedforward sdl constructions of output-buffered multiplexers and switches with variable length bursts," Proc. 26th IEEE INFOCOM 2007, pp.679–687, 2007.

[10] F. Jordan, D. Lee, K. Lee, and S.V. Ramanan, "Serial array time slot interchangers and optical implementations," IEEE Trans. Commun., vol.43, no.11, pp.1309–1318, 1994.

[11] D. Hunter and G. Smith, "New architectures for optical TDM switching," J. Lightware Technol., vol.11, no.3, pp.495–511, 1993.

Xiaoliang Wang received his B.S., M.S. degrees in 2003 and 2006 respectively, all from Xidian University, Xi'an, China. He is currently pursing the Ph.D. degree in the Graduate School of Information Sciences, Tohoku University, Japan. His research interests include optical switching networks, scheduling algorithm and router design.

Xiaohong Jiang received his B.S., M.S. and Ph.D. degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xi'an, China. He is currently an Associate Professor in the Department of Computer Science, Graduate School of Information Science, TOHOKU University, Japan. Before joining TOHOKU University, Dr. Jiang was an assistant professor in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), from Oct. 2001 to Jan. 2005. Dr. Jiang was a JSPS (Japan Society for the Promotion of Science) postdoctoral research fellow at JAIST from Oct. 1999–Oct. 2001. He was a research associate in the Department of Electronics and Electrical Engineering, the University of Edinburgh from March 1999–Oct. 1999. Dr. Jiang's research interests include optical switching networks, routers, network coding, WDM networks, VoIP, interconnection networks, IC yield modeling, timing analysis of digital circuits, clock distribution and fault-tolerant technologies for VLSI/WSI. He has published over 120 referred technical papers in these areas. He is a member of IEEE.

Susumu Horiguchi (M'81-SM'95) received the B.Eng the M.Eng and PhD degrees from Tohoku University in 1976, 1978 and 1981 respectively. He is currently a Full Professor in the Graduate School of Information Sciences, Tohoku University. He was a visiting scientist at the IBM Thomas J. Watson Research Center from 1986 to 1987. He was also a professor in the Graduate School of Information Science, JAIST (Japan Advanced Institute of Science and Technology). He has been involved in organizing international workshops, symposia and conferences sponsored by the IEEE, IEICE, IASTED and IPS. He has published over 150 papers technical papers on optical networks, interconnection networks, parallel algorithms, high performance computer architectures and VLSI/WSI architectures. Prof. Horiguchi is members of IPS and IASTED.