Construction of Subexponential-Size Optical Priority Queues With Switches and Fiber Delay Lines

Bin Tang, Member, IEEE, ACM, Xiaoliang Wang, Member, IEEE, Cam-Tu Nguyen, Member, IEEE, Baoliu Ye, Member, IEEE, and Sanglu Lu, Member, IEEE

Abstract—All-optical switching has been considered as a natural choice to keep pace with growing fiber link capacity. One key research issue of all-optical switching is the design of optical buffers for packet contention resolution. One of the most general buffering schemes is optical priority queue, where every packet is associated with a unique priority upon its arrival and departs the queue in order of priority, and the packet with the lowest priority is always dropped when a new packet arrives but the buffer is full. In this paper, we focus on the feedback construction of an optical priority queue with a single \((M+2) \times (M+2)\) optical crossbar Switch and \(M\) fiber Delay Lines (SDL) connecting \(M\) inputs and \(M\) outputs of the switch. We propose a novel construction of an optical priority queue with buffer \(2^{O(\sqrt{M^3})}\), which improves substantially over all previous constructions that only have buffers of \(O(M^c)\) size for constant integer \(c\). The key ideas behind our construction include (i) the use of first in first out multiplexers, which admit efficient SDL constructions, for feeding back packets to the switch instead of fiber delay lines, and (ii) the use of a routing policy that is similar to self-routing, where each packet entering the switch is routed to some multiplexer mainly determined by the current ranking of its priority.

Index Terms—Optical priority queue, optical switch, fiber delay lines, optical multiplexer.

I. INTRODUCTION

All-optical packet switching is very attractive for making a good use of the enormous bandwidth of optical networks, since it eliminates the complicated and quite expensive optical-electrical-optical conversions. One main issue for implementing all-optical packet switching is the construction of optical buffers for conflict resolutions among packets competing for the same resources. As optical-RAM is not available yet, a common approach for constructing optical buffers is to use a combination of bufferless optical crossbar Switches and fiber Delay Lines (SDLs), where fiber delay lines (FDLs) act as storage devices for optical packets [2]–[5]. However, unlike the traditional electronic memories with random access, one packet entering an FDL must propagate for a fixed amount of time and cannot be retrieved anytime earlier. Such inflexibility makes the design of SDL-based optical buffers with the same throughput and delay performance as its electronic counterpart quite challenging. In the past one decade and a half, great efforts have been made on constructing various kinds of optical buffers, such as first in first out (FIFO) multiplexers [6]–[12], FIFO queues [13]–[16], last in first out (LIFO) queues [15], [17], [18], priority queues [19]–[24], and shared queues [25], [26], etc.

In this paper, we focus on the design of optical priority queues with SDLs. A priority queue contains an arrival link, a departure link, and a loss link. Each packet is associated with a unique priority upon its arrival. When a departure request is raised by a controller, the packet with the highest priority is sent out from the departure link. If a new packet arrives but the buffer of the priority queue is full, then the packet with the lowest priority is dropped via the loss link. Priority queue is one of the most general buffering schemes, as the priority of each packet can be assigned arbitrarily. In particular, both FIFO queues and LIFO queues can be viewed as priority queues where the arrival time of a packet is used as its priority.

Following previous works [19]–[21], [24], we consider the construction of an optical priority queue using a feedback system as illustrated in Fig. 1. This system consists of an \((M+2) \times (M+2)\) optical crossbar switch, which has one distinguished input for external packet arriving, one distinguished output for packet departure, one distinguished output for packet loss, and \(M\) FDLs with delays \(d_1, d_2, \ldots, d_M\) connecting the other inputs and outputs in pairs. The issue is to choose proper delays \(d_1, d_2, \ldots, d_M\) as well as the routing policy performed by the switch, such that the switching system can exactly emulate a priority queue.

All the arrival time and priorities of packets and the packet departure requests can be arbitrary, making the optical priority queue highly dynamic. This leads to the design of delays of FDLs and the routing policy in a coupled way very difficult. In particular, there are two basic necessary conditions for the routing policy:

- **Delay condition:** a packet with the \(i\)-th highest priority cannot be switched into an FDL with delay higher than \(i\).


Digital Object Identifier 10.1109/TNET.2019.2960402
Based on these conditions, Sarwate and Anatharam [19] showed that the buffer size is upper bounded by $2^M + 1$.

To accommodate the conditions, they introduced a routing policy based on sorting the priorities of the packets entering the switch. Proper delays were further assigned to the FDLs, which leads to the first construction of optical priority queue with buffer $\Theta(M^2)$ [19]. This sorting-based routing policy plays a vital role in all the subsequent constructions of optical priority queues, including the ones by Chiu et al. in [20] and [21] whose buffer sizes are $\Theta(M^2)$ and $\Theta(M^3)$, respectively, and the recursive construction by Datta [24], which can achieve a buffer size of $\Theta(M^c)$ for any positive integer $c$. However, all these buffer sizes achieved are polynomial in $M$, which are far away from the exponential upper bound $2^M + 1$.

In this paper, we make a great step towards closing the above gap by presenting a novel construction of an optical priority queue with buffer $2^{\Theta(\sqrt{M})}$. To the best of our knowledge, this is the first construction of an optical priority queue whose buffer size goes beyond polynomials of the number of FDLs $M$. The key ideas behind our construction include two aspects.

- **Collision-free condition**, i.e., for any time and any FDL, there must be at most one packet entering the FDL. Based on these conditions, Sarwate and Anatharam [19] showed that the buffer size is upper bounded by $2^M + 1$.

- We introduce a novel routing policy that is similar to self-routing [6], where each packet entering the switch is routed to some multiplexer mainly determined by the current ranking of its priority according to a simple routing rule. Compared to the sorting-based routing policy used in all previous constructions, our routing policy also incurs a lower computation cost.

Specifically, we adopt 4-to-1 multiplexers and use them in groups each of which consists of three same 4-to-1 multiplexers. By using an exponential sequence for setting the buffer sizes of multiplexers and an appropriate routing rule, we can guarantee that neither packet collision nor buffer overflow could happen at each multiplexer. Based on these salient properties, we show that our construction emulates a priority queue exactly. Although our construction uses multiple switches, we can combine all the switches into one, and finally have a construction of an optical priority queue with buffer $2^{\Theta(\sqrt{M})}$ using a single crossbar switch and $M$ fiber delay lines.

The remainder of this paper is organized as follows. In Sec. II, we introduce the basic assumptions and definitions used throughout this paper. In Sec. III, we present a very
efficient construction of optical priority queues while the proof is given in Sec. IV. Sec. V discusses about related work. Finally, Sec. VI presents the concluding remarks.

II. PRELIMINARIES

In this section, we first introduce the basic assumptions and network elements adopted in this paper and then introduce the definition of priority queue.

A. Assumptions and Basic Network Elements

As in most work about the SDL-based optical queue designs [6]–[26], we assume that the time of system is slotted and synchronized, and the packet size is fixed such that one packet can be transmitted over a link within one time slot. Since there is at most one packet in a link, we can use 0-1 variables to characterize the state of a link. We say that a link is in state 1 at time \(t\) if there is a packet in the link at \(t\), and the link is in state 0 at \(t\) otherwise.

Switches and fiber delay lines are defined as follows.

Definition 1 (Switch): An \(n \times n\) (optical) crossbar switch is a memoryless network element that has \(n\) input links and \(n\) output links, which can realize all the \(n!\) permutations between its inputs and outputs. Specifically, for any \(k\) such that \(k \leq n\), packets coming from any \(k\) input links will instantaneously go out from \(k\) output links which are specified by a protocol performed by the switch. We will refer to \(n\) as the size of the switch and the protocol as the routing policy of the switch.

Definition 2 (Fiber delay line, FDL): A fiber delay line with delay \(d\) (a non-negative integer) is a network element that has one input link and one output link, through which \(d\) time slots are required for a packet to traverse. Let \(a(t)\) denote the state of the input link at time \(t\). Then the state of the output link at \(t\) is \(a(t-d)\).

When a packet is traversing through an FDL, it looks like that the packet is buffered in the FDL. Therefore, an FDL can be viewed as a memory device, but it is much more inflexible than traditional electronic memory since at most one packet can enter the FDL at one time slot and a packet entering the the FDL can only be retrieved after a fixed amount of time.

B. Priority Queues

Consider the network element shown in Fig. 3, which has an input link for packet arrival, one controller, and two output links, one for departing packets, and the other for loss packets. Every packet arriving at the network element is associated with a unique label, called priority, which is used to indicate the expected departure order of this packet among all the buffered packets. Suppose there are \(k\) packets at the beginning of time \(t\), including the arriving packet if any, in the switching system. If a packet \(i\) has the \(j\)-th highest priority among the \(k\) packets, we say that \(i\) has a tag of \(j\) at time \(t\), which is denoted by \(\tau_i(t) = j\). Hence, a packet having a smaller tag has a higher priority than a packet having a larger tag at any time. However, the tag of a packet buffered in the system can change over time due to the arrival and departure of other packets.

We use the following notations to describe the state of the network element at each time \(t\).

- Let \(a(t)\), \(d(t)\) and \(l(t)\) denote the states of the input link, the departure link and the loss link at time \(t\), respectively.
- Let \(c(t) = 1\) if the controller sends a departure request at time \(t\) and \(c(t) = 0\) otherwise.
- Denote by \(q(t)\) the number of packets buffered in the network element at time \(t\).

A discrete-time priority queue can then be defined formally as follows.

Definition 3 (Priority Queue): Starting empty at time 0, the network element shown in Fig. 3 is called a priority queue with buffer \(B\) if it satisfies all the following properties at each time \(t > 0\):

1. **Flow conservation**: arriving packets are either stored in the network element or transmitted through the departure link or the loss link, i.e.,
   \[ q(t) = q(t-1) + a(t) - d(t) - l(t) \] (1)

2. **Non-idling**: If there are packets buffered in the network element or there is an arriving packet, then there is a packet departing from the network element if and only if the controller sends a departure request, i.e.,
   \[ d(t) = \begin{cases} 
   1 & \text{if } c(t) = 1 \text{ and } q(t-1) + a(t) > 0 \\
   0 & \text{otherwise.} 
\end{cases} \] (2)

3. **Maximum buffer usage**: There is a packet dropped out from the loss link if and only if there is no departure request, the buffer is full and there is an arriving packet, i.e.,
   \[ l(t) = \begin{cases} 
   1 & \text{if } c(t) = 0, q(t-1) = B \text{ and } a(t) = 1 \\
   0 & \text{otherwise.} 
\end{cases} \] (3)

4. **Priority departure**: If there is a departure packet \(i\) at time \(t\), then \(i\) must have the highest priority among all the packets buffered in the network element and the arriving packet (if any) at time \(t\), i.e.,
   \[ \tau_i(t) = 1. \] (4)

5. **Priority loss**: If there is a loss packet \(i\) at time \(t\), then \(i\) much have the lowest priority among all the \(B\) packets buffered in the network element and the arriving packet at time \(t\), i.e.,
   \[ \tau_i(t) = B + 1. \] (5)
The number of packets buffered at the multiplexer at time

In this paper, we focus on the construction of optical priority queues with a single optical crossbar switch and M FDLs as shown in Fig. 1. The efficiency of a construction is evaluated by the buffer size of the constructed optical priority queue in terms of M.

C. Multiplexers

Our construction of optical priority queue will use FIFO multiplexers as intermediate building blocks. For the sake of completeness, we give a formal definition of multiplexers.

Definition 4 (Multiplexer): An n-to-1 (FIFO) multiplexer with buffer B is a network element with n input links, one departure link, and n − 1 output links for packet losses. Let \( \hat{a}_i(t) \), \( i = 1, 2, \ldots, n \), be the state of the i-th input link, \( \hat{d}(t) \) be the state of the departure link and \( \hat{l}_i(t) \), \( i = 1, 2, \ldots, n − 1 \), be the state of the i-th loss link, and \( \hat{q}(t) \) be the number of packets buffered at the multiplexer at time t. The n-to-1 multiplexer with buffer B satisfies the following four properties.

(M1) Flow conservation: arriving packets from the n input links are either stored in the buffer or transmitted through the n output links, i.e.,

\[
\hat{q}(t) = \hat{q}(t - 1) + \sum_{i=1}^{n} \hat{a}_i(t) - \hat{d}(t) - \sum_{i=1}^{n-1} \hat{l}_i(t). \tag{6}
\]

(M2) Non-idling: there is always a departing packet if there are packets in the buffer or there are arriving packets, i.e.,

\[
\hat{d}(t) = \begin{cases} 
1 & \text{if } \hat{q}(t - 1) + \sum_{i=1}^{n} \hat{a}_i(t) > 0 \\
0 & \text{otherwise}.
\end{cases} \tag{7}
\]

(M3) Maximum buffer usage: arriving packets are lost only when the buffer is full, i.e., for \( i = 1, \ldots, n - 1 \),

\[
\hat{l}_i(t) = \begin{cases} 
1 & \text{if } \hat{q}(t - 1) + \sum_{i=1}^{n} \hat{a}_i(t) \geq B + i + 1 \\
0 & \text{otherwise}.
\end{cases} \tag{8}
\]

(M4) FIFO: packets depart in the FIFO order.

See Fig. 4 for an illustration of a 4-to-1 multiplexer with buffer B. As also mentioned in Sec. I, a multiplexer with buffer B is much more flexible than an FDL with delay B.

Specifically,

- A multiplexer has multiple inputs, which brings extra room for the design of routing policy as the collision-free condition is easier to satisfy.

- The buffer of a multiplexer is always used in a consecutive manner, so it can be fully utilized, and as long as the number of packets buffered does not exceed the buffer size, there would never be any buffer overflow. On the other hand, it is very hard to fully use an FDL viewed as a buffer. See Fig. 5 for an illustration.

III. CONSTRUCTION OF OPTICAL PRIORIT Y QUEUES

In this section, we present a very efficient construction of optical priority queues based on multiplexers, and analyze its construction cost in terms of SDLs.

To ease the presentation, we introduce some notations regarding sets of consecutive integers. Let \( \Psi \) be a set of consecutive integers. Define \( L(\Psi) \) and \( U(\Psi) \) be the smallest integer and the largest integer in \( \Psi \), respectively. That is, \( \Psi = \{L(\Psi), L(\Psi) + 1, \ldots, U(\Psi)\} \). For simplicity, we write \( \Psi = (L(\Psi), U(\Psi)) \).

In order to help understand our construction, we start by introducing the motivation behind our design idea.

A. Motivation

Consider the construction of an optical priority queue using a feedback system as illustrated in Fig. 1, and suppose that there are \( M = 2^\ell - 1 \) FDLs indexed by 1, 2, \ldots, M for some positive integer \( \ell \). One necessary condition for the design of delays of FDLs and the routing policy is that, a packet with the i-th highest priority cannot be switched into an FDL with delay higher than i. Otherwise, if there is a departure request while no packet arrives in each of the next \( i \) time slots, the packet with the i-th highest priority cannot leave the system in time.

One basic idea to satisfy the above condition is that, set the delays of FDLs as 1, 2, 4, \ldots, \( 2^{\ell-2} \), \( 2^{\ell-1} \), \( 2^{\ell-2} \), \ldots, 4, 2, 1, and use a self-routing policy as follows: let packet with tag belonging to \( \Psi_j \) enter FDL j, where for \( j = 1, 2, \ldots, \ell \),

\[
\Psi_j = (2^{j-1}, 2^j - 1),
\]

and for \( j = \ell + 1, \ell + 2, \ldots, 2\ell - 1 \),

\[
\Psi_j = (3 \times 2^{\ell-1} - 2^{\ell-j}, 3 \times 2^{\ell-1} - 2^{\ell-j} - 1).
\]

The third column of Table I gives the values of \( \Psi_j \) for \( \ell = 5 \). (Here the delay sequence and the tag set sequence exhibit a symmetric structure which is employed for the priority loss property.) This setting is “ideal” in the sense that the switching system can buffer up to \( O(2^\ell) \) packets. However, this setting fails to be a priority queue. The underlying issue is collision, i.e., there will be multiple packets with tags belonging to a
same $\Psi_j$ that enter a same FDL at the same time according to the routing policy.

As multiplexers have multiple inputs providing the possibility to solve the collision issue, we are motivated to replace each FDL with a multiplexer with buffer equal to the delay of the FDL. However, this cannot solve the collision issue completely since the number of packets entering a multiplexer can be larger than the number of inputs of the multiplexer (which should be a limited number for construction efficiency). Besides, we need to get rid of buffer overflow at each multiplexer.

To solve the collision issue fundamentally, our key idea is to use multiple multiplexers with smaller buffers as a group to replace each FDL instead of using a single multiplexer. In this way, we can guarantee that the packets entering a group of multiplexers can only come from certain groups of multiplexers except for the arrival link, which have a limited number. So by using multiplexers with a proper number of inputs, the collisions can be avoided. Also, we can establish an upper bound on the number of packets that need to be buffered at some group of multiplexers, and then choose a proper number of multiplexers in a group such that the total buffer size exceeds the upper bound. Thanks to the property that the buffer of a multiplexer is always used in a consecutive manner as mentioned in Sec. II, buffer overflow can thus never happen at each multiplexer as long as the buffers of the multiplexers in a same group are equally used (differing by at most one packet).

B. Description of the Construction

Now we formally introduce our construction of optical priority queue.

1) Structure: Let $\ell$ be a positive integer. In our construction, an optical priority queue, as illustrated in Fig. 6, consists of a $(24\ell-10) \times (24\ell-10)$ crossbar switch and $2\ell-1$ groups of multiplexers. For each $j = 1, 2, \ldots, 2\ell-1$, the $j$-th group of multiplexers consists of three parallel 4-to-1 multiplexers with buffer $B_j$, where

$$B_j = \begin{cases} 1 & j = 1 \\ 2^{j-2} & j = 2, 3, \ldots, \ell \\ 2^{2\ell-j-2} & j = \ell + 1, \ell + 2, \ldots, 2\ell - 2 \\ 1 & j = 2\ell - 1. \end{cases}$$

So each group of multiplexers has 12 input links in total. For $i = 0, 1, 2$, we label the 4 input links in the $i$-th multiplexer as $i$-th, $(i+3)$-th, $(i+6)$-th and $(i+9)$-th input links of the group of multiplexers. See Fig. 7 for an illustration. The reason for using three multiplexers each with four inputs in a group will be clear after our analysis (c.f. Remark 3 and Lemma 10).

Recall the definition of $\Psi_j$ given in Sec. III-A. We have

$$|\Psi_j| = \begin{cases} B_j & j = 1 \text{ or } j = 2\ell - 1 \\ 2B_j & j = 2, 3, \ldots, 2\ell - 2. \end{cases}$$

Let

$$B^* \triangleq 3 \times 2^{\ell-1} - 2 = U(\Psi_{2\ell-1}).$$

Table I gives an example on these parameters where $\ell = 5$.

2) Routing Policy: The routing policy performed by the switch at the beginning of time $t$, $t = 1, 2, \ldots$, is as follows.

- If there is a departure request, i.e., $c(t) = 1$, then compare the packets out from the first and the second groups of multiplexers as well as the arriving packet, if any, and let the one having the highest priority depart this switching system from the departure link.
- If there is no departure request and there is an arriving packet, but the buffer is full, i.e., $c(t) = 0$, $a(t) = 1$ and $q(t-1) = B^*$, then compare the arriving packet and the packets out from the last group of multiplexers, if any, and let the packet having the lowest priority leave the switching system from the loss link.

![Fig. 6. The construction of an optical priority queue based on 4-to-1 multiplexers. Each group of 4-to-1 multiplexers consists of three 4-to-1 multiplexers with same buffer size. Here the loss links of 4-to-1 multiplexers are omitted.](image)

![Fig. 7. A group of 4-to-1 multiplexers consists of three 4-to-1 multiplexers. The indices of 12 input links are labelled. Here the loss links are omitted.](image)
TABLE I
SOME PARAMETERS OF AN OPTICAL CONSTRUCTION OF PRIORITY QUEUES WHERE \( \ell = 5 \) AND \( B^* = 46 \). THE COLUMN OF TAG RANGE IS DUE TO LEMMA 7, AND THE LAST COLUMN IS DUE TO LEMMA 10

| \( j \) | \( B_j \) | \( \Psi_j \) | tag range | num. of buffered pkt |
|-------|---------|-----------|-----------|---------------------|
| 1     | 1       | \{1\}     | (1)       | \( \leq 1 \)        |
| 2     | 1       | (2, 3)    | (2, 3)    | \( \leq 2 \)        |
| 3     | 2       | (4, 7)    | (3, 8)    | \( \leq 5 \)        |
| 4     | 4       | (8, 15)   | (5, 18)   | \( \leq 11 \)       |
| 5     | 8       | (16, 31)  | (9, 38)   | \( \leq 23 \)       |
| 6     | 4       | (32, 39)  | (29, 42)  | \( \leq 11 \)       |
| 7     | 2       | (40, 43)  | (39, 44)  | \( \leq 5 \)        |
| 8     | 1       | (44, 45)  | (44, 45)  | \( \leq 2 \)        |
| 9     | 1       | \{46\}    | \{46\}    | \( \leq 1 \)        |

- Every other packet \( i \) entering the switch will be pushed into the \( j \)-th group of multiplexer such that

\[
\tau_i(t) \in \Psi_j. \tag{9}
\]

Specifically, suppose that there are \( k \) packets entering the \( j \)-th group of multiplexer according to (9), and let \( u_j(t) \) be the index of the input link of the group that is lastly used before \( t \), where \( u_j(1) = 0 \). Then these \( k \) packets will enter the \( j \)-th group via \((u_j(t)+1)\%12\)-th, \((u_j(t)+2)\%12\)-th, ..., \((u_j(t)+k)\%12\)-th input links, respectively. Also, set \( u_j(t+1) = (u_j(t)+k)\%12 \). Here \( x\%y \) denotes the remainder of \( x \) when divided by \( y \).

It is straightforward to see that, the computation cost of the above routing policy at each time is linear with the number of packets entering the switch at that time, or equivalently, \( O(\ell) \). It is remarkable that this routing policy is feasible only if there is no packet collision at each group of multiplexer, i.e., the number of packets entering a group of multiplexers each time is at most 12, the total number of input links of the group of multiplexers. As we will show in Lemma 8, this requirement can always be satisfied.

We have the following main result.

**Theorem 1:** The proposed switching system is an optical priority queue with buffer \( B^* \).

The proof of Theorem 1 is deferred to Sec. IV.

### C. Construction Cost

The cost of our construction of optical priority queues depends on how to construct 4-to-1 multiplexers with SDLs. As will be demonstrated in Lemma 12 in Sec. IV, there would never be any buffer overflow at each multiplexer. Based on this fact, some requirements on the used 4-to-1 multiplexers could be relaxed.

- First, any 4-to-1 multiplexer with \( B_j \) buffer could be replaced by a 4-to-1 multiplexer with buffer larger than or equal to \( B_j \). This is because, when no buffer overflow happens at either of them, they have identical departure processes if both are started from empty systems and subject to identical arrival processes.

- Second, a 4-to-1 multiplexer could be replaced by a 4-to-1 delayed-loss multiplexer with the same buffer size.

A 4-to-1 delayed-loss multiplexer and a 4-to-1 multiplexer with the same buffer size have identical departure processes if they are started from empty systems and subject to identical arrival processes. The only difference is that the loss processes of the two systems do not match exactly. See [6] for a formal definition of delayed-loss multiplexers.

Regarding the construction of delayed-loss multiplexers, we have the following result based on the construction proposed by Chang et al. [6].

**Lemma 2:** For any positive integer \( k \), an \( n \)-to-1 delayed-loss multiplexer with buffer \( n^k - 1 \) can be constructed with a \(((n-1)k+n)\times((n-1)k+n)\) crossbar switch and \((n-1)k\) FDLs.

**Proof:** Chang et al. [6] gave a construction of an \( n \)-to-1 delayed-loss multiplexer with buffer \( n^k - 1 \), which consists of \( k+1 \) \( n \times n \) crossbar switches, indexed by 0, 1, ..., \( k \), in tandem. More specifically, the \( n \) input links of the \( 0 \)-th crossbar switch act as the \( n \) input links of the \( n \)-to-1 delayed-loss multiplexer, and the \( n \) output links of the \( k \)-th crossbar switch act as the output link and \( n - 1 \) loss links of the multiplexer. For \( i = 0, 1, \ldots, k - 1 \), the \( n \) output links of the \( i \)-th crossbar switch connect to \( n \) input links of the \( i+1 \)-th crossbar switch, each via an FDL with some specific delay except one via a direct link. See [6, Fig. 17] for an illustration. This construction uses \((n-1)k\) FDLs in total.

Now consider the integration of all the switches in the construction into one. A straightforward integration will consist of \((nk+n)\times(nk+n)\) crossbar switch, \((n-1)k\) FDLs which connect \((n-1)k\) outputs and \((n-1)k\) inputs of the switch, and \( k \) direct links which connect \( k \) outputs and \( k \) inputs of the switch. Note that the \( k \) direct links become useless in this integration. So the links together with the corresponding inputs and outputs can be removed from this integration, which leads to a construction of \(((n-1)k+n)\times((n-1)k+n)\) crossbar switch and \((n-1)k\) FDLs.

Define

\[
B'_j = \begin{cases} 
3 & j = 1 \\
4^{\ell-1} - 1 & j = 2, 3, \ldots, \ell \\
4^{\ell-2} - 1 & j = \ell + 1, \ell + 2, \ldots, 2\ell - 2 \\
3 & j = 2\ell - 1.
\end{cases}
\]

It is straightforward to check that \( B'_j \geq B_j \) for all \( j = 1, 2, \ldots, 2\ell - 1 \).

In order to take advantage of Lemma 2, we replace each 4-to-1 multiplexer with buffer \( B_j \), \( j = 1, 2, \ldots, 2\ell - 1 \) in our construction with a 4-to-1 delayed-loss multiplexer with buffer \( B'_j \). We refer to this construction as specialized construction. According to our analysis, the specialized construction is also an optical priority queue with buffer \( B^* \).

By further integrating all the switches used into one, we have the following result (the result for the trivial case that \( \ell = 1 \) is omitted).

**Theorem 3:** For any positive integer \( \ell \geq 2 \), an optical priority queue with \( 3 \times 2^{\ell-1} - 2 \) can be constructed with a \((\frac{3}{2}(9\ell^2 + 39\ell) + 8) \times (\frac{1}{2}(9\ell^2 + 39\ell) + 8)\) crossbar switch and \( 3(\ell^2 - \ell) + 18 \) FDLs.
Proof: We consider the specialized construction where we adopt the method in Lemma 2 to construct the 4-to-1 delayed-loss multiplexers. According to Lemma 2, the number of FDLs used by this construction is

\[ 3 \left( 3 + \sum_{j=2}^{\ell} \left( \frac{j-1}{2} + 1 \right) + 2^{-2} \sum_{j=\ell+1}^{2\ell-2} \left( \frac{2\ell-j-1}{2} + 3 \right) + 3 \right) \]

\[ = \frac{9}{2} (\ell^2 - \ell) + 18, \]

which holds for both \( \ell \) is even and \( \ell \) is odd.

Note that when integrating all the switches into one, the size of the integrated switch is the sum of the sizes of all the switches used for constructing 4-to-1 delayed-loss multiplexers plus 2 other than plus 24\( \ell \)−10. According to Lemma 2, this is equal to

\[ 3 \left( 7 + \sum_{j=2}^{\ell} \left( 3 \left( \frac{j-1}{2} + 4 \right) \right) + 2^{-2} \sum_{j=\ell+1}^{2\ell-2} \left( 3 \left( \frac{2\ell-j-1}{2} + 4 \right) + 7 \right) + 2 \right) \]

\[ = \frac{1}{2} (9\ell^2 + 39\ell) + 8. \]

The proof is accomplished.

Considering the construction framework depicted in Fig. 1, we have the following result.

Theorem 4: There exists a construction of optical priority queue with buffer 2\( \Theta(\sqrt{M}) \) using a single (\( M + 2 \)) \times (\( M + 2 \)) crossbar switch and \( M \) FDLs.

Proof: Let \( M = \frac{1}{2} (9\ell^2 + 39\ell) + 6 \). Then \( \ell = \Theta(\sqrt{M}) \) and \( B^e = 3 \times 2^{\ell-1} - 2 = 2^{\Theta(\sqrt{M})} \). According to Theorem 3, this result holds directly.

Remark 1: It remains open that how to construct a 4-to-1 multiplexer with an arbitrary size optimally in the sense of the number of FDLs used. Any more efficient method for constructing 4-to-1 multiplexers with buffer size \( B_3 \) than the one for constructing a 4-to-1 delayed-loss multiplexers with buffer size \( B'_3 \) would lead to better results than Theorem 3.

Remark 2: The construction cost given in Theorem 3 can be reduced by, e.g., replacing the first/last group of multiplexers by a single FDL, replacing each multiplexer in the second/last second group by a single FDL, etc. But these changes can only reduce the construction cost by a small fixed number, which does not change the result in the order sense.

IV. PROOF OF THEOREM 1

According to Definition 3, we need to show that the proposed switching system satisfies all the properties (P1)-(P5) for any time \( t > 0 \). We will prove this by induction on time \( t \). It is straightforward to check that these properties hold in the base case \( t = 1 \). For induction, we assume that the proposed switching system satisfies all the properties (P1)-(P5) for every time \( t < T \). We will show that it also satisfies these properties for \( t = T \).

The proof proceeds as follows. First, we will give some basic results about the changing of the tag of a packet. Then, we will prove that the proposed routing policy is collision-free, which guarantees the feasibility of the routing policy. Later, we will show that there is no buffer overflow at each multiplexer. Finally, we will prove Theorem 1 based on the preparations.

A. Tag Changing

We first show that the tag of any packet in the switching system can change by at most one in a time slot.

Lemma 5: For any packet \( i \) in this switching system at both time \( t \) and time \( t - 1 \) where \( t \leq T \),

\[ |\tau_i(t) - \tau_i(t-1)| \leq 1, \quad (10) \]

Proof: Since the switching system emulates a priority queue up to time \( T - 1 \), we can show the result based on the properties of a priority queue. We consider two cases:

Case 1: there is no arriving packet at time \( t \). As (P1)-(P5) hold for time \( t - 1 \), it is straightforward to see that \( \tau_i(t) = \tau_i(t-1) - 1 \) if there exists a departure packet at time \( t - 1 \), or \( \tau_i(t) = \tau_i(t-1) - 1 \) if otherwise. Hence, (10) holds.

Case 2: there is an arriving packet at time \( t \). If the packet has a lower priority than \( i \), then the argument for case 1 also holds. If the packet has a higher priority than \( i \), then \( \tau_i(t) = \tau_i(t-1) - 1 \) if there exists a departure packet at time \( t - 1 \), or \( \tau_i(t) = \tau_i(t-1) + 1 \) if otherwise. For all these subcases, (10) holds.

Lemma 5 directly implies the following result, which is a generalization of Lemma 5.

Corollary 6: For any packet \( i \) in the switching system at both time \( t \) and time \( t' \) where \( t' < t \leq T \),

\[ |\tau_i(t) - \tau_i(t')| \leq t - t'. \]

B. Collision-Free

We first show the range of the tag of a packet buffered at some group of multiplexers.

Lemma 7: For any packet \( i \) buffered at the \( j \)-th group of multiplexers at time \( t < T \),

\[ L(\Psi_j) - B_j + 1 \leq \tau_i(t) \leq L(\Psi_j) + B_j - 1. \]

Proof: Consider a packet \( i \) buffered at some multiplexer in the \( j \)-th group of multiplexers at time \( t < T \). Let \( t' \leq t \) be time that \( i \) entered the multiplexer for the last time. According to properties (M2) and (M4) of multiplexers, \( i \) would depart from the multiplexer in at most \( B_j \) time steps since \( t' \). Hence, \( t - t' \leq B_j - 1 \). By Corollary 6, we have

\[ |\tau_i(t) - \tau_i(t')| \leq t - t' \leq B_j - 1. \]

This completes the proof as \( \tau_i(t') \in \Psi_j \) according to the routing policy.

The following result shows that the proposed routing policy is collision-free, which guarantees the feasibility of the proposed routing policy.

Lemma 8: For any \( j \), the number of packets entering the \( j \)-th group of multiplexers at time \( T \) under the routing policy is at most 10.
Let these multiplexers have the same number of inputs. Hence, to 10. Meanwhile, in order to guarantee the buffers of multi-

The proof is accomplished.

In the following, we will show that there would not be any buffer overflow at each multiplexer. We start by showing that the difference between the tags of any pair of packets in the switching system can change by at most 1 in a time slot.

Lemma 9: For any $t \leq T$ and any packets $i_1$ and $i_2$ in this switching system at both time $t$ and time $t-1$,

\[ |(\tau_{i_1}(t) - \tau_{i_2}(t)) - (\tau_{i_1}(t-1) - \tau_{i_2}(t-1))| \leq 1. \] (11)

Proof: We consider all the four possible cases:

Case 1: there is no packet arriving at the switching system at $t$. If there exists a departure packet at time $t-1$, then $\tau_{i_1}(t) = \tau_{i_1}(t-1) - 1$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1) - 1$. Otherwise, $\tau_{i_1}(t) = \tau_{i_1}(t-1)$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1)$. For both of the subcases, (11) holds.

Case 2: there is a packet arriving at the switching system at $t$, which has a lower priority than both $i_1$ and $i_2$. Clearly, the argument for case 1 also holds.

Case 3: there is a packet arriving at the switching system at $t$, which has a higher priority than both $i_1$ and $i_2$. If there exists a departure packet at time $t-1$, then $\tau_{i_1}(t) = \tau_{i_1}(t-1) - 1$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1) - 1$. Otherwise, $\tau_{i_1}(t) = \tau_{i_1}(t-1) + 1$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1) + 1$. Hence, (11) holds.

Case 4: there is a packet arriving at the switching system at $t$, which has a higher priority than $i_1$ but lower than $i_2$ (without loss of generality, we here assume $i_1$ has a lower priority than $i_2$). If there exists a departure packet at time $t-1$, then $\tau_{i_1}(t) = \tau_{i_1}(t-1) - 1$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1) - 1$. Otherwise, $\tau_{i_1}(t) = \tau_{i_1}(t-1) + 1$ and $\tau_{i_2}(t) = \tau_{i_2}(t-1) + 1$. Hence, (11) also holds in this case.

By Corollary 6, we also have

\[ \left| \tau_{i_1}(t) - \tau_{i_2}(t) \right| \leq 3B_j - 2. \] (12)

Proof: Suppose that the time that packets $i_1$ and $i_2$ entered the $j$-th group of multiplexers for the last time before $T$ or at time $T$ is $t_1$ and $t_2$. Without loss of generality, we assume that $t_2 \leq t_1$. By Lemma 9, we have

\[ \left| \tau_{i_1}(t) - \tau_{i_2}(t) \right| \leq \left| \tau_{i_1}(t_1) - \tau_{i_2}(t_1) \right| + (T - t_1). \] (13)

Combining (12), (13) and (14), we have

\[ \left| \tau_{i_1}(T) - \tau_{i_2}(T) \right| \leq T - t_2 + 2B_j - 1. \]

Note that $T - t_2 \leq B_j - 1$ since otherwise $i_2$ would leave the $j$-th group of multiplexers before $T$. This completes the proof.
For $t < T$, let $q_j(i, t), i = 0, 1, 2$ be the number of packets buffered at the $i$-th multiplexer in the $j$-th group of multiplexers at time $t$, and let $q_j(i, T), i = 0, 1, 2$ be the number of packets buffered at or entering the $i$-th multiplexer in the $j$-th group of multiplexers at time $T$. Recall that, according to the routing policy, the input links of a group of multiplexers are used in a round-robin manner. Based on this scheme and together with the non-idling property (M2) of multiplexers, we can show that the buffers of the multiplexers in a same group are always almost equally used, i.e., the number of packets buffering in the multiplexers differs by at most one. Besides, if the input link that is lastly used before $t$ belongs to the $i$-th multiplexer in the group, then at time $t - 1$, the number of packets buffered in the $i$-th multiplexer at time $t - 1$ is equal to or larger than that in the $((i - 1)\%3)$-th multiplexer, which is also equal to or larger than that in the $((i - 2)\%3)$-th multiplexer.

**Lemma 11:** Consider any $j$-th group of multiplexers, and let $k(t) = u_j(t + 1)\%3$. Then, for $t \leq T$,

$$q_j(k(t), t) \geq q_j((k(t) - 1)\%3, t) \geq q_j((k(t) - 2)\%3, t),$$

(15)

and

$$q_j(k(t), t) - q_j((k(t) - 2)\%3, t) \leq 1.$$  

(16)

**Proof:** We prove this result by induction on $t$. If $t = 0$, then $k(t) = 0$, and $q_j(0, 0) = q_j(1, 0) = q_j(2, 0) = 0$. So, (15) and (16) hold.

Now suppose (15) and (16) hold when $t = t_0 - 1 < T - 1$.

We will show that they also hold for $t = t_0$. Without loss of generality, we assume that $k(t_0 - 1) = 0$ (the cases that $k(t_0 - 1) = 1$ and $k(t_0 - 1) = 2$ can be considered in a same way). By induction hypothesis, we have

$$q_j(0, t_0 - 1) \geq q_j(2, t_0 - 1) \geq q_j(1, t_0 - 1)$$

(17)

and

$$q_j(0, t_0 - 1) - q_j(1, t_0 - 1) \leq 1.$$  

(18)

Let $m(i, t_0), i = 0, 1, 2$, denote the number of packets arriving at the $i$-th multiplexer in the group at time $t_0$. Then,

$$k(t_0) = u_j(t_0 + 1)\%3$$

$$= \left( u_j(t_0) + \sum_{i=0}^{2} m(i, t_0) \right) \%3$$

$$= \left( \frac{2}{3} m(i, t_0) \right) \%3$$

where the second equality holds according to the routing policy and the last equality holds since $u_j(t_0)\%3 = k(t_0 - 1) = 0$.

Since the routing policy uses the multiplexers in a same group in a round robin manner, we have

$$m(k(t_0), t_0) \geq m((k(t_0) - 1)\%3, t_0) \geq m((k(t_0) - 2)\%3, t_0).$$

(19)

and

$$m(k(t_0), t_0) - m((k(t_0) - 2)\%3, t_0) \leq 1.$$  

(20)

Since the switching system emulates the priority queue for each $t < T$, there is no packet lost at time $t_0$ if $t_0 < T$.

Hence, according to (M1) and (M2),

$$q_j(i, t_0) = [q_j(i, t_0 - 1) + m(i, t_0) - 1]^+, \quad i = 0, 1, 2.$$  

(21)

Clearly, the above equality also holds for $t_0 = T$ due to the definition of $q_j(i, T)$. We consider three possible cases:

**Case 1:** $\sum_{i=0}^{2} m(i, t_0)\%3 = 0$. Then $k(t_0) = 0$. By (19) and (20), $m(0, t_0) = m(1, t_0) = m(2, t_0)$. From (17), (18) and (21), it is straightforward to see that $q_j(0, t_0) \geq q_j(2, t_0)$ and $q_j(0, t_0) - q_j(1, t_0) \leq 1$.

**Case 2:** $\sum_{i=0}^{2} m(i, t_0)\%3 = 1$. Then $k(t_0) = 1$. By (19) and (20), $m(1, t_0) = m(0, t_0) + 1 = m(2, t_0) + 1$, and, from (17), (18) and (21), we can get $q_j(1, t_0) \geq q_j(0, t_0) \geq q_j(2, t_0)$ and $q_j(1, t_0) - q_j(2, t_0) \leq 1$.

**Case 3:** $\sum_{i=0}^{2} m(i, t_0)\%3 = 2$. Then $k(t_0) = 2$. By (19) and (20), $m(1, t_0) = m(2, t_0) = m(0, t_0) + 1$, and $k(t_0) = 2$. From (17), (18) and (21), we have $q_j(2, t_0) \geq q_j(1, t_0) \geq q_j(0, t_0) + q_j(2, t_0) - q_j(0, t_0) \leq 1$.

Hence, for each case, we have (15) and (16) for $t = t_0$.

By mathematical induction, (15) and (16) hold for $t \leq T$.

**Lemma 12:** Any packet arriving at any multiplexer in the switching system at time $T$ cannot be lost.

**Proof:** By contradiction, we assume that there exists some packet arriving at $i$-th multiplexer in the $j$-th group lost due to overflow. According to (M3), $q_j(i, T) > B_j$. By Lemma 11, this implies that $q_j(i', T) \geq B_j$ for $i' \neq i \in \{1, 2, 3\}$. Hence, $q_j(1, T) + q_j(2, T) + q_j(3, T) > 3B_j$. On the other hand, Lemma 10 implies $q_j(1, T) + q_j(2, T) + q_j(3, T) \leq 3B_j - 1$, which leads to a contradiction. The proof is accomplished. 

**D. Completing the Proof**

Now we complete the proof of Theorem 1. We will show that all the five properties (P1)-(P5) hold at time $T$. First, according to Lemma 8 and Lemma 12, (P1) holds directly.

To prove (P2) and (P4), we can assume, without loss of generality, that $c(T) = 1$ and $q(T - 1) + a(T) > 0$. Consider the packet $i$ that $\tau_i(T) = 1$. If it is the arriving packet, then according to the routing policy, (P2) and (P4) hold directly. If otherwise, $\tau_i(T - 1) = 1$ or $\tau_i(T - 1) = 2$ according to Lemma 5. By Lemma 7, we can check that $i$ must be buffered at the first group of multiplexers or at the second group of multiplexers at time $T - 1$. Recall that the buffer size of each multiplexer in the first group or in the second group is just one. By property (M2), packet $i$ will leave the corresponding multiplexer and enter the switch at time $T$. Hence, according to the routing policy, (P2) and (P4) hold.

(P3) and (P5) can be proved similar to (P2) and (P4). Suppose that there is no departure request and there is an arriving packet at time $T$, while the number of packets buffered in the switching system at time $T - 1$ is $B^*$. Consider the packet $i$ that $\tau_i(T) = B^* + 1$. If $i$ is the arriving packet, then according to the routing policy, it will be dropped via the loss link at $T$. Hence, (P3) and (P5) hold. If otherwise, then according to Lemma 5, $\tau_i(T - 1) = B^*$. By Lemma 7,
was buffered at the last group of multiplexers. Recall that the buffer size of each multiplexer in the last group is just one. By property (M2), packet \( i \) will leave the corresponding multiplexer at time \( T \). According to the routing policy, \( i \) will be dropped via the loss link at \( T \). Hence, (P3) and (P5) hold at \( T \).

The whole proof is accomplished.

V. RELATED WORK

Many methods have been developed for using the SDL-based constructions to exactly emulate various electronic queue structures. Here we introduce the constructions of some typical SDL-based optical components.

- **FIFO multiplexers**: In [4], a design named COD (Cascaded Optical Delay-lines) was proposed for exactly emulating 2-to-1 FIFO multiplexers by using \( 2 \times 2 \) crossbar switches and FDLs. However, the number of switches in COD is linear in the buffer size. An improved design named Logarithm Delay-Line Switched was proposed in [27] where the number of \( 2 \times 2 \) switches used is only logarithmic in the buffer size. In [6], a recursive construction of 2-to-1 multiplexer was introduced, which was further extended to constructing \( n \)-to-1 multiplexers using self-routing. In [8], it was proposed that an \((M + 2) \times (M + 2)\) crossbar switch and \( M = O(\log B)\) FDLs are sufficient to emulate a 2-to-1 multiplexer with buffer \( B \). Based on these works, some other constraints including fault-tolerance [9], variable length burst [7], and limited number of recirculations [11] are also taken into account for constructing 2-to-1 multiplexers. Since FIFO multiplexers admit efficient SDL based constructions and have some salient properties that FDLs do not have, they can be exploited in the design of optical priority queues, as firstly demonstrated in this work.

- **FIFO and LIFO queues**: In [13], a recursive construction for FIFO queue was proposed which uses \( 2 \log_2 B - 1 \) FDLs, where \( B \) is the buffer size. In [16], a necessary and sufficient condition was characterized for SDL constructions of FIFO queues. In [17], a cascade optical LIFO queue architecture based on multiple building-block modules was developed, but its capacity of each module is highly limited. In [15], the idea of two-level caching was proposed, based on which recursive constructions of parallel FIFO and LIFO queues are proposed. The result in [15] indicate that a LIFO queue of size \( B \) can be constructed using at most \( 9 \log_2 B \) FDLs. An improved design was proposed in [18], which only uses approximately \( 3 \log_2 B \) FDLs. Although FIFO and LIFO queues can be viewed as special cases of priority queues, existing ideas for constructing FIFO and LIFO queues cannot be easily extended for constructing priority queues.

- **Priority queues**: In [19], Sarwate and Anatharam firstly considered the SDL-based construction of optical priority queues. They showed the buffer size is upper bounded by \( 2^M + 1 \), where \( M \) is the number of FDLs, and gave a construction of an optical priority queue with \( \Theta(M^2) \) buffer. A more general construction framework based on the notion of complementary priority queue was proposed in [20]. Using this framework, an improved design of optical priority queue with \( \Theta(M^2) \) buffer was proposed in [21]. These results were extended to the construction of optical priority queues with multiple inputs and multiple outputs in [23]. Very recently, a recursive construction of optical priority queue was proposed which can achieve a buffer size of \( \Theta(M^{c}) \) for any positive integer \( c \). All these constructions considered the exact emulation of optical priority queues. In contrast, “strong” emulation of optical priority queue was considered in [22] where each packet departs from the construction with bounded delay.

VI. CONCLUSION

We have proposed a novel construction of an optical priority queue with buffer \( 2^\Theta(\sqrt{M}) \) using a single optical crossbar switch and \( M \) FDLs, which leverages 4-to-1 multiplexers for feeding back packets to the switch, and adopts a routing policy that is similar to self-routing. This is a substantial improvement over all previous constructions of optical priority queues which only have polynomial-size buffers. In the future, we would make further efforts towards closing the remaining gap between the exponential upper bound in [19] and the established sub-exponential lower bound for the SDL design of priority queues. We would also like to see whether our method can be extended to achieve better designs of other network elements (e.g., optical priority queues with multiple inputs and multiple outputs [23]).

REFERENCES

[1] B. Tang, X. Wang, C.-T. Nguyen, and S. Lu, “Constructing sub-exponentially large optical priority queues with switches and fiber delay lines,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jul. 2016, pp. 1441–1445.
[2] M. J. Karol, “Shared-memory optical packet (ATM) switch,” Multigigabit Fiber Commun. Syst., vol. 2024, pp. 212–223, Nov. 1993.
[3] I. Chlamtac et al., “CORD: Contention resolution by delay lines,” INEE Sel. Areas Commun., vol. 14, no. 5, pp. 1014–1029, Jun. 1996.
[4] R. L. Cruz and I.-T. Tsai, “COD: Alternative architectures for high speed packet switching,” IEEE/ACM Trans. Netw., vol. 4, no. 1, pp. 11–21, Feb. 1996.
[5] D. K. Hunter, W. D. Cornwell, T. H. Gilfedder, A. Franzen, and I. Andonovic, “SLOB: A switch with large optical buffers for packet switching,” J. Lightw. Technol., vol. 16, no. 10, pp. 1725–1736, Oct. 1998.
[6] C.-S. Chang, D.-S. Lee, and C.-K. Tu, “Recursive construction of FIFO optical multiplexers with switched delay lines,” IEEE Trans. Inf. Theory, vol. 50, no. 12, pp. 3221–3233, Dec. 2004.
[7] C.-S. Chang, D.-S. Lee, and C.-K. Tu, “Using switched delay lines for exact emulation of FIFO multiplexers with variable length bursts,” IEEE J. Sel. Areas Commun., vol. 24, no. 4, pp. 108–117, Apr. 2006.
[8] 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, Oct. 2006.
[9] J. Cheng, “Constructions of fault-tolerant optical 2-to-1 FIFO multiplexers,” IEEE Trans. Inf. Theory, vol. 53, no. 11, pp. 4002–4105, Nov. 2007.
[10] Y.-T. Chen, C.-S. Chang, J. Cheng, D.-S. Lee, and C.-C. Huang, “Feedforward SDL constructions of output-buffered multiplexers and switches with variable length bursts,” in Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM), May 2007, pp. 679–687.
[11] J. Cheng, C.-S. Chang, T.-H. Chao, D.-S. Lee, and C.-M. Lien, “On constructions of optical queues with a limited number of recirculations,” in Proc. 27th IEEE Int. Conf. Comput. Commun. (INFOCOM), Apr. 2008, pp. 664–672.
[12] J. Cheng, C.-S. Chang, S.-H. Yang, T.-H. Chao, D.-S. Lee, and C.-M. Lien, “Greedy constructions of optical queues with a limited number of recirculations,” IEEE Trans. Inf. Theory, vol. 63, no. 8, pp. 5314–5326, Aug. 2017.
[13] C.-S. Chang, Y.-T. Chen, and D.-S. Lee, “Constructions of optical FIFO queues,” IEEE Trans. Inf. Theory, vol. 52, no. 6, pp. 2838–2843, Jun. 2006.
[14] S.-Y. R. Li and X. J. Tan, “Mux/demux queues, FIFO queues, and their construction by fiber memories,” IEEE Trans. Inf. Theory, vol. 57, no. 3, pp. 1328–1343, Mar. 2011.
[15] P. K. Huang, C. S. Chang, J. Cheng, and D. S. Lee, “Recursive constructions of parallel FIFO and LIFO queues with switched delay lines,” IEEE Trans. Inf. Theory, vol. 53, no. 5, pp. 1778–1798, May 2007.
[16] J. Cheng, H.-H. Chou, and C.-H. Cheng, “A necessary and sufficient condition for SDL constructions of optical FIFO queues,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2013, pp. 2339–2345.
[17] B. A. Small, A. Shacham, and K. Bergman, “A modular, scalable, extensible, and transparent optical packet buffer,” J. Lightw. Technol., vol. 25, no. 4, pp. 978–985, Apr. 2007.
[18] X. Wang, X. Jiang, and A. Pattavina, “Efficient designs of optical LIFO buffer with switches and fiber delay lines,” IEEE Trans. Commun., vol. 59, no. 12, pp. 3430–3439, Dec. 2011.
[19] A. D. Sarwate and V. Anantharam, “Exact emulation of a priority queue with a switch and delay lines,” Queueing Syst., Theory Appl., vol. 53, no. 3, pp. 115–125, Jul. 2006.
[20] H.-C. Chiu, C.-S. Chang, J. Cheng, and D.-S. Lee, “A simple proof for the constructions of optical priority queues,” Queueing Syst., Theory Appl., vol. 56, no. 2, pp. 73–77, Jun. 2007.
[21] H.-C. Chiu, C.-S. Chang, J. Cheng, and D.-S. Lee, “Using a single switch with $O(M^2)$ inputs/outputs for the construction of an optical priority queue with $O(M^3)$ buffer,” in Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM), May 2007, pp. 2501–2505.
[22] H. Kogan and I. Keslassy, “Optimal-complexity optical router,” in Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM), May 2007, pp. 706–714.
[23] J. Cheng, H.-C. Chiu, C.-S. Chang, and D.-S. Lee, “Constructions of optical priority queues with multiple inputs and multiple outputs,” IEEE Trans. Inf. Theory, vol. 57, no. 7, pp. 4274–4301, Jul. 2011.
[24] A. Datta, “Construction of polynomial-size optical priority queues using linear switches and fiber delay lines,” IEEE/ACM Trans. Netw., vol. 25, no. 2, pp. 974–987, Apr. 2017.
[25] X. Wang, X. Jiang, A. Pattavina, and S. Horiguchi, “A construction of 1-to-2 shared optical buffer queue with switched delay lines,” IEEE Trans. Commun., vol. 57, no. 12, pp. 3712–3723, Dec. 2009.
[26] X. Wang, X. Jiang, and A. Pattavina, “Constructing $N$-to-$N$ shared optical queues with switches and fiber delay lines,” IEEE Trans. Inf. Theory, vol. 58, no. 6, pp. 3836–3842, Jun. 2012.
[27] D. K. Hunter, D. Cotter, R. B. Ahmad, and W. D. Cornwell, “2 × 2 buffered switch fabrics for traffic routing, merging, and shaping in photonic cell networks,” J. Lightw. Technol., vol. 15, no. 1, pp. 86–101, Jan. 1997.