

SDL Constructions of FIFO, LIFO and Absolute Contractors

Cheng-Shang Chang, Jay Cheng, Duan-Shin Lee
Institute of Communications Engineering
National Tsing Hua University
Hsinchu 30013, Taiwan, R.O.C.
Email: cschang@ee.nthu.edu.tw
jcheng@ee.nthu.edu.tw
lds@cs.nthu.edu.tw

Abstract—Despite all the recent advances in the mathematical theories for constructing optical queues by optical Switches and fiber Delay Lines (SDL), there are still many problems that need to be resolved. In this paper, we tackle the following problems: (i) is it possible to construct optical queues with switches of arbitrary sizes? (ii) is there a general theory that unifies many constructions of optical queues with known packet delays? and (iii) under what conditions can a concatenation of optical queues allow overtaking? For the first problem, we propose a new class of optical memory cells that can be made by switches of arbitrary sizes. Moreover, we propose the generalized \mathcal{C} -transform for routing packets through such optical memory cells. For the second problem, we introduce a new class of optical queues, including FIFO, LIFO and absolute contractors. We show that both linear compressors in [13] and FIFO multiplexers (with multiple inputs) in [5], [7] are special cases of contractors. An interesting finding is that overtaking can occur in LIFO and absolute contractors.

I. INTRODUCTION

One of the critical challenges of all-optical packet switching is the lack of optical buffers. To cope with such a problem, there are many recent advances in the mathematical theories for constructing optical queues by optical Switches and fiber Delay Lines (SDL), e.g., First-In First-Out (FIFO) multiplexers in [1]–[2] and [5]–[7], buffered packet switches in [2]–[3], FIFO queues in [8]–[9], Last-In First-Out (LIFO) queues in [9], priority queues in [11]–[12], and linear compressors, non-overtaking delay lines, and flexible delay lines in [13].

Despite all the advances in the mathematical theories for constructing optical queues, there are still many problems that need to be resolved. The main contributions of this paper is to tackle some of these problems as listed below:

- (i) Many of the SDL constructions for optical queues in the literature are constructed by using only a set of 2×2 switches. Is it possible to extend the constructions to switches with arbitrary sizes?
- (ii) There are many constructions of optical queues with known packet delays, e.g., linear compressors in [13] and the FIFO multiplexers (with multiple inputs) in [5], [7]. Is there a general theory that unifies all these constructions?
- (iii) Concatenations of various optical queues usually do not allow overtaking. Under what conditions can

overtaking be allowed without causing any conflicts?

For the first problem, we first extend the scaled optical memory cell made by 2×2 switches in [13], [8] to a more general setting so that optical memory cells can be made by switches of arbitrary sizes. We then propose the generalized \mathcal{C} -transform for routing packets through such optical memory cells. The generalized \mathcal{C} -transform is a generalization of the \mathcal{C} -transform in [6] that goes beyond the usual binary representations.

For the second problem, we introduce a new class of optical queues, called *contractors*. As suggested by its name, a contractor realizes all the contraction mappings from interarrival times to interdeparture times. It is called a FIFO (resp. LIFO, absolute) contractor if the order of the mapping is preserved (resp. reversed, irrelevant). We show that both linear compressors in [13] and FIFO multiplexers (with multiple inputs) in [5], [7] are special cases of some forms of contractors.

The development of the theory for the absolute (and LIFO) contractors also sheds some light on the third problem as overtaking is allowed in these optical queues. Such results are quite surprising as the construction by concatenations of various optical memory cells is feedforward in nature.

This paper is organized as follows. In Section II, we first introduce new optical memory cells constructed by switches with various sizes. We then develop the generalized \mathcal{C} -transform for routing policies in a concatenation of optical memory cells. In Section III, we introduce the concept of FIFO contractors and develop theories for the constructions of such optical queues. We then extend FIFO contractors to absolute and LIFO contractors in Section IV. In Section V, we further extend the absolute contractor from one input to multiple inputs. The paper is concluded in Section VI, where we address possible extensions of our work.

II. BASIC CONSTRUCTION UNITS AND ROUTING POLICIES

A. Generalizations of optical memory cells

In this paper, we consider constructions of optical queues in the discrete-time setting, i.e., (i) packets are assumed to be of the same size, and (ii) time is slotted and synchronized so that a packet can be transmitted in one time slot.

One of the basic network elements used for constructing many optical queues in [8], [13] is the (scaled) optical memory

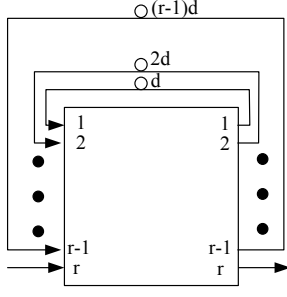


Fig. 1. An optical memory cell with parameters r and d .

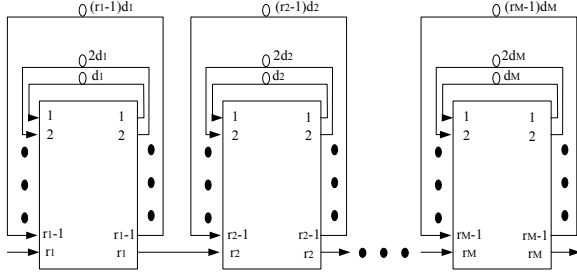


Fig. 2. A concatenation of optical memory cells with various parameters.

cell that consists of a 2×2 switch and a fiber delay line connected from one of its outputs to one of its inputs. In the following, we first extend the optical memory cell in [8], [13] to a more general setting so that optical memory cells can be made by switches of arbitrary sizes.

Definition 1: ((\mathbf{r}, \mathbf{d})-cell) As shown in Figure 1, an optical memory cell with parameters r and d , denoted by an (r, d) -cell, consists of an $r \times r$ optical crossbar switch and $r - 1$ fiber delay lines with delays $d, 2d, \dots, (r - 1)d$. These $r - 1$ fiber delay lines are connected from $r - 1$ outputs to $r - 1$ inputs of the $r \times r$ optical crossbar switch. The $r \times r$ optical crossbar switch is assumed to be capable of realizing all the $r!$ permutations from the r inputs to the r outputs.

Clearly, the optical memory cell with scaling factor d in [8], [13] is a $(2, d)$ -cell. With the generalization in Definition 1, the constructions of optical queues will not be limited to the usage of 2×2 switches as in [8], [13]. Moreover, we note that an (r, d) -cell is different from the RAM defined in [10], where all the fiber delay lines are of the same length.

In Figure 2, we show a concatenation of M optical memory cells with parameters $(r_1, d_1), (r_2, d_2), \dots, (r_M, d_M)$. A routing policy associated with such a construction is a way to set up the connection patterns of the M crossbar switches in such a construction for any time t . An optical queue is said to be realizable by such a construction if for all sample paths of the optical queue, there is a routing policy that can route packets through it without colliding with each other. As shown in [8], [13], many optical queues, including time slot interchange, FIFO queues, linear compressors, linear decompressors, and non-overtaking delay lines, can be realized by the concatenation of optical memory cells with $r_i = 2$, $i = 1, 2, \dots, M$, under appropriate routing policies.

Definition 2: (Loop-once policy) A routing policy for a concatenation of optical memory cells in Figure 2 is said to be a *loop-once* policy if every packet can be routed back from an optical memory cell at most once.

If a concatenation of optical memory cells in Figure 2 is operated under a loop-once policy, then there is no collision of packets in the delay lines with non-zero delays. This is because if there is a collision in the delay line of some optical memory cell with delay $d > 0$ in Figure 2, then there must be at least two packets at the inputs of that optical memory cell and at least one of them is routed back to the delay line with delay d . This violates the loop-once property. Thus, the only places that might have collisions in a concatenation of optical memory cells under a loop-once policy are *the links that connect the M optical memory cells*.

A routing policy for a concatenation of optical memory cells is said to be a *non-overtaking* policy if there is no overtaking of packets at each optical memory cell. Clearly, under a non-overtaking policy, there is no collision at the links that connect the M optical memory cells. This leads to the following proposition.

Proposition 3: If there is a loop-once and non-overtaking policy for a concatenation of optical memory cells in Figure 2, then there is no collision inside the construction.

B. Generalized \mathcal{C} -transform

The \mathcal{C} -transform in [6] is a very important method to generate a *binary* representation for a quantity x . In the following, we extend it to a more general setting that makes an r -ary presentation possible. This will then in turn lead to a loop-once policy in a concatenation of optical memory cells in Figure 2.

Definition 4: (Generalized \mathcal{C} -transform) Consider two M -vectors $\mathbf{r} = (r_1, r_2, \dots, r_M)$ and $\mathbf{d} = (d_1, d_2, \dots, d_M)$ with $r_i \in \{2, 3, 4, \dots\}$ and $d_i \in \{1, 2, 3, \dots\}$, $i = 1, 2, \dots, M$. Define a mapping $\mathcal{C}^g : \{0, 1, 2, \dots\} \mapsto \{0, 1, \dots, r_1 - 1\} \times \{0, 1, \dots, r_2 - 1\} \times \dots \times \{0, 1, \dots, r_M - 1\}$ as follows:

$$\mathcal{C}^g(x) = \left(I_1^g(x), I_2^g(x), \dots, I_M^g(x) \right), \quad (1)$$

where $I_M^g(x), I_{M-1}^g(x), \dots, I_1^g(x)$ (in that order) are given recursively by

$$I_i^g(x) = \begin{cases} r_i - 1, & \text{if } x - \sum_{k=i+1}^M I_k(x) \cdot d_k \geq (r_i - 1)d_i \\ j, & \text{if } jd_i \leq x - \sum_{k=i+1}^M I_k(x) \cdot d_k \\ & < (j + 1)d_i, \quad j = 1, 2, \dots, r_i - 2, \\ 0, & \text{if } x < d_i, \end{cases} \quad (2)$$

where we adopt the convention that the sum in (2) is zero if the upper index is smaller than its lower index.

We call $\mathcal{C}^g(x)$ the generalized \mathcal{C} -transform of x with respect to $\mathbf{r} = (r_1, r_2, \dots, r_M)$ and $\mathbf{d} = (d_1, d_2, \dots, d_M)$. The superscript g is used for differentiating the *generalized* \mathcal{C} -transform from the \mathcal{C} -transform in [6]. Note that if we choose $r_i = r$ and $d_i = r^{i-1}$, $i = 1, 2, \dots, M$, then the generalized \mathcal{C} -transform yields the usual r -ary representation.

Clearly, the C -transform in [6] is a special case of the generalized C -transform with $r_i = 2$, $i = 1, 2, \dots, M$. However, one can also map a generalized C -transform to the C -transform in [6] by enlarging the state space. To see this, let $N_0 = 0$, $N_k = \sum_{j=1}^k (r_j - 1)$, $k = 1, 2, \dots, M$, and consider the N_M -vector $\mathbf{v} = (v_1, v_2, \dots, v_{N_M})$ with

$$v_\ell = d_k, \text{ for } N_{k-1} + 1 \leq \ell \leq N_k \text{ and } 1 \leq k \leq M. \quad (3)$$

Let $\mathcal{C}(x) = (I_1(x), I_2(x), \dots, I_{N_M}(x))$ be the \mathcal{C} -transform with respect to the N_M -vector \mathbf{v} . Then for any $0 \leq x \leq S_M^g = \sum_{i=1}^M (r_i - 1) \cdot d_i$, it is easy to see from induction that for $k = M, M-1, \dots, 1$

$$I_k^g(x) = \sum_{\ell=N_{k-1}+1}^{N_k} I_\ell(x), \quad (4)$$

and

$$\begin{aligned} \sum_{k=i}^M I_k^g(x) \cdot d_k &= \sum_{k=i}^M \sum_{\ell=N_{k-1}+1}^{N_k} I_\ell(x) \cdot d_k \\ &= \sum_{\ell=N_{i-1}+1}^{N_M} I_\ell(x) \cdot \mathbf{v}_\ell. \end{aligned} \quad (5)$$

This is further illustrated in the following example.

Example 5: (Generalized C -transform) Suppose that $r_1 = 4, r_2 = 3$, and $d_1 = 1, d_2 = 3$. In this case, $S_M^g = (r_1 - 1)d_1 + (r_2 - 1)d_2 = 9$. With the enlarge state space in (3), we have $N_1 = 3, N_2 = 5$ and

$$\mathbf{v} = (d_1, d_1, d_1, d_2, d_2) = (1, 1, 1, 3, 3).$$

Then the generalized C -transform and the corresponding C -transform of x for $0 \leq x \leq 9$ are shown in Table I.

x	$I_1^g(x)$	$I_2^g(x)$	$I_1(x)$	$I_2(x)$	$I_3(x)$	$I_4(x)$	$I_5(x)$
0	0	0	0	0	0	0	0
1	1	0	0	0	1	0	0
2	2	0	0	1	1	0	0
3	0	1	0	0	0	0	1
4	1	1	0	0	1	0	1
5	2	1	0	1	1	0	1
6	0	2	0	0	0	1	1
7	1	2	0	0	1	1	1
8	2	2	0	1	1	1	1
9	3	2	1	1	1	1	1

TABLE I

THE GENERALIZED C -TRANSFORM AND THE CORRESPONDING C -TRANSFORM OF x WITH RESPECT TO $r_1 = 4, r_2 = 3$, AND $d_1 = 1, d_2 = 3$.

In view of (4) and (5), we have the following results for the generalized C -transform immediately from Lemmas 4 and 5, and Corollary 6 in [6].

Proposition 6: Consider the generalized C -transform with respect to $\mathbf{r} = (r_1, r_2, \dots, r_M)$ and $\mathbf{d} = (d_1, d_2, \dots, d_M)$ in Definition 4. Suppose that $x, y \in \{0, 1, 2, \dots\}$ and $1 \leq i \leq M$.

$$(i) \quad 0 \leq \sum_{k=i}^M I_k^g(x) \cdot d_k \leq x.$$

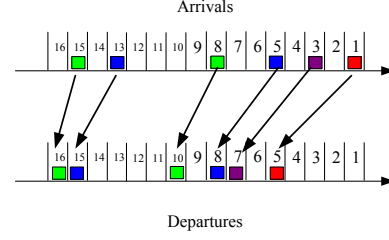


Fig. 3. A sample path of a FIFO contractor.

(ii) **(Uniqueness)**

$$\begin{aligned} \sum_{k=i}^M I_k^g(x) \cdot d_k &= \sum_{k=i}^M I_k^g(y) \cdot d_k \text{ if and only if} \\ I_k^g(x) &= I_k^g(y), \quad k = i, i+1, \dots, M. \end{aligned}$$

(iii) **(Monotonicity)** If $0 \leq x \leq y$, then

$$\sum_{k=i}^M I_k^g(x) \cdot d_k \leq \sum_{k=i}^M I_k^g(y) \cdot d_k.$$

(iv) If $x = \sum_{k=i}^M I_k^g(y) \cdot d_k$ for some y , then $x = \sum_{k=i}^M I_k^g(x) \cdot d_k$.

(v) **(Complete decomposition)** Assume that $d_1 = 1$, and $1 \leq d_{i+1} \leq \sum_{k=1}^i (r_k - 1) \cdot d_k + 1$, $i = 1, 2, \dots, M-1$. Then

$$x = \sum_{k=1}^M I_k^g(x) \cdot d_k \text{ for all } x \text{ with } 0 \leq x \leq S_M^g,$$

where $S_M^g = \sum_{k=1}^M (r_k - 1) \cdot d_k$.

III. FIFO CONTRACTORS

In this section, we show that the generalized C -transform can be used for generating a loop-once and non-overtaking policy in a concatenation of optical memory cells for a FIFO contractor defined below.

Definition 7: (FIFO contractor) A FIFO contractor is a network element with one input link and one output link. Let $\tau^a(n)$ be the arrival time of the n^{th} packet at the input link and $\tau^d(n)$ be the departure time of the n^{th} packet at the output link. Suppose that the departure time of a packet is known upon its arrival. A *FIFO contractor* with the range of delay $[d_1, d_2]$ realizes the set of mappings (or sample paths) that satisfy

$$\tau^a(n) + d_1 \leq \tau^d(n) \leq \tau^a(n) + d_2, \quad \text{for all } n, \quad (6)$$

and the following condition:

$$0 < \tau^d(n) - \tau^d(n-1) \leq \tau^a(n) - \tau^a(n-1) \quad (7)$$

whenever $\tau^a(n) \leq \tau^d(n-1)$. In particular, if $d_1 = 0$, then it is called a FIFO contractor with maximum delay d_2 .

The condition in (7) implies that the interdeparture times are not larger than the interarrival times for those packets in the same busy period, i.e.,

$$0 < \tau^d(n) - \tau^d(m) \leq \tau^a(n) - \tau^a(m) \quad (8)$$

for any two packets (with packet indices $m < n$) in the same busy period. As such, if we view each sample path as a mapping from arrivals to departures, then the mapping is a *contraction mapping* from interarrival times to interdeparture times (as shown in Figure 3).

Another view of a FIFO contractor is to rewrite (7) as follows:

$$\tau^d(n) - \tau^a(n) \leq \tau^d(n-1) - \tau^a(n-1). \quad (9)$$

Thus, the delays for those packets in the same busy period are decreasing in the packet index.

Note that $\tau^a(n) - \tau^a(n-1) \geq 1$ as there is at most one packet arrival in a time slot in the input link. The condition in (7) is weaker than

$$\tau^d(n) = \tau^d(n-1) + 1$$

in a linear compressor in [13]. As such, a FIFO contractor can realize a larger set of sample paths than that of a linear compressor. As in a linear compressor, the first packet that initiates a busy period in a FIFO contractor can have an arbitrary delay as long as the delay is within the specified range. Then the delays of the subsequent packets in that period are *decreasing* in the packet index.

A. A construction by a concatenation of optical memory cells

In this section, we will show that a FIFO contractor can be constructed by a concatenation of optical memory cells in Figure 2. Moreover, such a construction permits self-routing of packets. Specifically, we assume that the i^{th} optical memory cell is with parameters r_i and d_i , $i = 1, 2, \dots, M$. For any integer $0 \leq x \leq S_M^g = \sum_{i=1}^M (r_i - 1) \cdot d_i$, let

$$\mathcal{C}^g(x) = \left(I_1^g(x), I_2^g(x), \dots, I_{M-1}^g(x), I_M^g(x) \right)$$

be the generalized \mathcal{C} -transform of x with respect to (r_1, r_2, \dots, r_M) and (d_1, d_2, \dots, d_M) . For a packet with delay x , it will be routed to the delay line with delay $j d_i$ in the i^{th} optical memory cell if $I_i^g(x) = j$. Clearly, such a routing policy is a loop-once policy. Also, we note that a packet arriving at time t and requesting a delay x will appear at the output link of the i^{th} memory cell at time $t + \sum_{k=1}^i I_k^g(x) \cdot d_k$, $i = 1, 2, \dots, M$.

Theorem 8: Under the above self-routing policy, the concatenation of optical memory cells in Figure 2 is a FIFO contractor with maximum delay S_M^g if d_i 's satisfy the complete decomposition property in Proposition 6(v), i.e., $d_1 = 1$, and $1 \leq d_{i+1} \leq \sum_{k=1}^i (r_k - 1) \cdot d_k + 1$, $i = 1, 2, \dots, M-1$.

Proof. Since the departure time of a packet in a FIFO contractor is known upon its arrival, its delay is also known. From the self-routing policy, a packet arriving at time t and requesting a delay x will appear at the output link of the M^{th} memory cell at time $t + \sum_{k=1}^M I_k^g(x) \cdot d_k$. As we assume that $d_1 = 1$ and $1 \leq d_{i+1} \leq \sum_{k=1}^i (r_k - 1) \cdot d_k + 1$, $i = 1, 2, \dots, M-1$, it then follows from the complete decomposition property in

Proposition 6(v) that

$$x = \sum_{k=1}^M I_k^g(x) \cdot d_k$$

for all $0 \leq x \leq S_M^g$. Thus, we achieve the exact delay of that packet.

It remains to show that there is no collision. Since the self-routing policy is a loop-once policy, from Proposition 3 we only need to show that it is also a non-overtaking policy. For this, let us consider two packets in the same busy period. Suppose that the n_1^{th} (resp. n_2^{th}) packet arrives at time t_1 (resp. t_2) with delay x_1 (resp. x_2). Without loss of generality, assume that $n_1 < n_2$ and $t_1 < t_2$. Note from the self-routing policy that packet n_1 (resp. n_2) will arrive at the output link of the i^{th} optical memory cell at $t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k$ (resp. $t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k$). Our objective is to show that there is no overtaking at the output link of each optical memory cell, i.e.,

$$t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k > t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k,$$

for all $i = 1, 2, \dots, M$.

From the complete decomposition property in Proposition 6(v), we have that

$$x_1 = \sum_{k=1}^M I_k^g(x_1) \cdot d_k, \quad (10)$$

and

$$x_2 = \sum_{k=1}^M I_k^g(x_2) \cdot d_k. \quad (11)$$

Thus,

$$\begin{aligned} & t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k - \left(t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k \right) \\ &= t_2 + x_2 - \sum_{k=i+1}^M I_k^g(x_2) \cdot d_k \\ & \quad - \left(t_1 + x_1 - \sum_{k=i+1}^M I_k^g(x_1) \cdot d_k \right) \\ &= t_2 + x_2 - (t_1 + x_1) \\ & \quad + \left(\sum_{k=i+1}^M I_k^g(x_1) \cdot d_k - \sum_{k=i+1}^M I_k^g(x_2) \cdot d_k \right). \quad (12) \end{aligned}$$

Since $t_1 + x_1$ (resp. $t_2 + x_2$) is simply the departure time of the n_1^{th} (resp. n_2^{th}) packet. From the first inequality in (8), we know that

$$t_2 + x_2 - (t_1 + x_1) > 0. \quad (13)$$

Also, in view of (9), we have $x_1 \geq x_2$. From the monotonicity result in Proposition 6(iii), it then follows that

$$\sum_{k=i+1}^M I_k^g(x_1) \cdot d_k \geq \sum_{k=i+1}^M I_k^g(x_2) \cdot d_k. \quad (14)$$

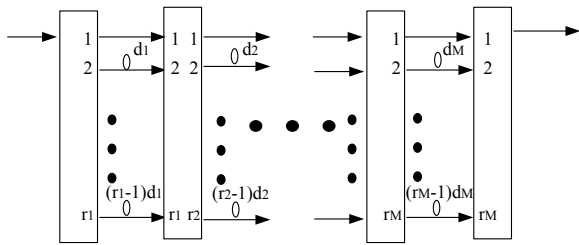


Fig. 4. A feedforward construction of a FIFO contractor.

Using (13) and (14) in (12) yields

$$t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k - \left(t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k \right) > 0.$$

The proof is completed. \blacksquare

B. A multistage feedforward construction

In Figure 4, we consider a multistage feedforward construction by optical crossbar switches and fiber delay lines. We will argue that such a construction can also be operated as a FIFO contractor.

There are $M + 1$ stages in the construction in Figure 4. In each stage, there is a crossbar switch with fiber delay lines connected to the next stage. The first stage consists of a $1 \times r_1$ crossbar switch and r_1 fiber delay lines with delays $0, d_1, 2d_1, \dots, (r_1 - 1)d_1$. The i^{th} stage, $i = 2, \dots, M$, consists of a $r_{i-1} \times r_i$ crossbar switch and r_i fiber delay lines with delays $0, d_i, 2d_i, \dots, (r_i - 1)d_i$. The $(M + 1)^{\text{th}}$ stage consists of an $r_M \times 1$ crossbar switch and one output link.

As in the self-routing policy in Section III-A, let

$$\mathcal{C}^g(x) = \left(I_1^g(x), I_2^g(x), \dots, I_{M-1}^g(x), I_M^g(x) \right)$$

be the generalized \mathcal{C} -transform of x with respect to (r_1, r_2, \dots, r_M) and (d_1, d_2, \dots, d_M) . For a packet with delay x , it will be routed to the delay line with delay $j d_i$ in the i^{th} stage if $I_i^g(x) = j$.

Under such a self-routing policy, we note that a packet arriving at time t and requesting a delay x will appear at one of the r_i input links of the $(i + 1)^{\text{th}}$ stage at time $t + \sum_{k=1}^i I_k^g(x) \cdot d_k$, $i = 1, 2, \dots, M$. As this is also the time that the packet arrives at the output link of the i^{th} optical memory cell in Figure 2. Thus, it follows from Theorem 8 that the construction in Figure 4 is also a FIFO contractor with maximum delay S_M^g if d_i 's satisfy the complete decomposition property in Proposition 6(v), i.e., $d_1 = 1$, and $1 \leq d_{i+1} \leq \sum_{k=1}^i (r_k - 1)d_k + 1$, $i = 1, 2, \dots, M - 1$. Moreover, there is at most one packet arriving at an optical crossbar switch at any time. Such a property is known as the crosstalk-free property in the literature (see e.g., [14]–[19]).

C. FIFO de-contractors

By reversing the direction of every optical link, we plot in Figure 5 the mirror image of the concatenation of optical

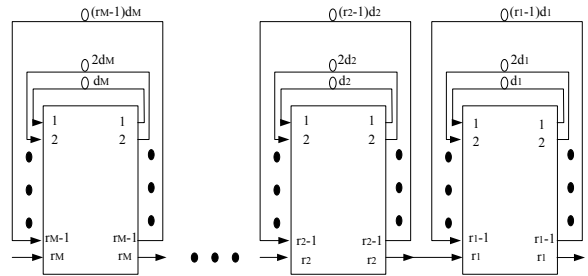


Fig. 5. The mirror image of the concatenation of optical memory cells in Figure 2.

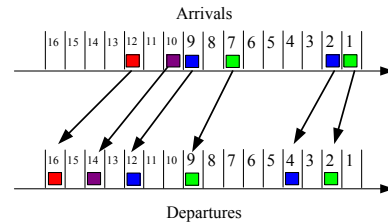


Fig. 6. A sample path of a FIFO de-contractor.

memory cells in Figure 2. It is known in [13] that the mirror image can realize every time reversed sample path in the original construction. As such, the construction in Figure 5 can be used as a FIFO de-contractor defined below.

Definition 9: (FIFO de-contractor) A FIFO de-contractor is defined exactly the same as a FIFO contractor in Definition 7 except the second inequality in (7) is reversed, i.e.,

$$\tau^d(n) - \tau^d(n-1) \geq \tau^a(n) - \tau^a(n-1) \quad (15)$$

whenever $\tau^a(n) \leq \tau^d(n-1)$.

Note that the condition in (15) implies that the interdeparture times are not less than the interarrival times for those packets in the same busy period, i.e.,

$$\tau^d(n) - \tau^d(m) \geq \tau^a(n) - \tau^a(m) \quad (16)$$

for any two packets (with packet indices $m < n$) in the same busy period (as shown in Figure 6).

Equivalently, one may rewrite (15) as follows:

$$\tau^d(n) - \tau^a(n) \geq \tau^d(n-1) - \tau^a(n-1). \quad (17)$$

As such, the delays of the packets in a busy period are *increasing* in a FIFO de-contractor, while they are *decreasing* in a FIFO contractor. Also, as the delays of packets in a busy period are *increasing* in a FIFO de-contractor, the FIFO condition is satisfied automatically, and a FIFO de-contractor realizes a much larger set of sample paths than that of a linear decompressor in [13].

Analogous to the self-routing rule for a FIFO contractor, we also use the generalized \mathcal{C} -transform for routing a packet in Figure 5. For a packet with delay x , it will be routed to the delay line with delay $j d_i$ in the $(M - i + 1)^{\text{th}}$ optical memory cell if $I_i^g(x) = j$. From such a self-routing policy, we note that a packet arriving at time t and requesting a delay

x will appear at the output of the i^{th} memory cell at time $t + \sum_{k=M-i+1}^M I_k^g(x) \cdot d_k$, $i = 1, 2, \dots, M$.

Theorem 10: Under the above self-routing policy, the concatenation of optical memory cells in Figure 5 is a FIFO de-contractor with maximum delay S_M^g if d_i 's satisfy the complete decomposition property in Proposition 6(v), i.e., $d_1 = 1$, and $1 \leq d_{i+1} \leq \sum_{k=1}^i (r_k - 1) \cdot d_k + 1$, $i = 1, 2, \dots, M - 1$.

Though Theorem 10 can be proved using the mirror image argument, we give a direct proof here.

Proof. Clearly, the self-routing policy based on the generalized \mathcal{C} -transform is a loop-once policy. Analogous to the proof of Theorem 8, we only need to show that it is also a non-overtaking policy. For this, let us consider two packets in the same busy period. Suppose that the n_1^{th} (resp. n_2^{th}) packet arrives at time t_1 (resp. t_2) with delay x_1 (resp. x_2). Without loss of generality, assume that $n_1 < n_2$ and $t_1 < t_2$. Note from the self-routing policy that packet n_1 (resp. n_2) will arrive at the output link of the i^{th} optical memory cell at $t_1 + \sum_{k=M-i+1}^M I_k^g(x_1) \cdot d_k$ (resp. $t_2 + \sum_{k=M-i+1}^M I_k^g(x_2) \cdot d_k$). Since $x_2 \geq x_1$ in (17), we have from the monotonicity result in Proposition 6(iii) that

$$\sum_{k=M-i+1}^M I_k^g(x_2) \cdot d_k \geq \sum_{k=M-i+1}^M I_k^g(x_1) \cdot d_k. \quad (18)$$

In conjunction with $t_1 < t_2$, we then have

$$t_2 + \sum_{k=M-i+1}^M I_k^g(x_2) \cdot d_k > t_1 + \sum_{k=M-i+1}^M I_k^g(x_1) \cdot d_k.$$

IV. ABSOLUTE CONTRACTORS

We have shown that the generalized \mathcal{C} -transform can be used for generating loop-once and non-overtaking policies in a concatenation of optical memory cells for both FIFO contractors and FIFO de-contractors. A natural question is then whether the generalized \mathcal{C} -transform can also be used to generate an *overtaking* policy for a certain optical queue that will not have any collisions in a concatenation of optical memory cells. The answer is positive as illustrated by the absolute contractor and the LIFO contractor defined below.

A. Definitions

Definition 11: (Absolute contractor) An absolute contractor is defined exactly the same as a FIFO contractor in Definition 7 except the condition in (7) is replaced by

$$0 < |\tau^d(n) - \tau^d(n-1)| \leq \tau^a(n) - \tau^a(n-1), \quad (19)$$

whenever $\tau^a(n) \leq \max_{j < n} \tau^d(j)$.

Note that if $\tau^a(n) > \max_{j < n} \tau^d(j)$, then there is no packet inside a network when the n^{th} packet arrives. Thus, the condition $\tau^a(n) \leq \max_{j < n} \tau^d(j)$ simply indicates that the

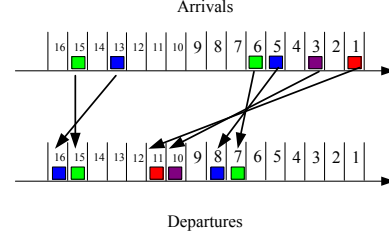


Fig. 7. A sample path of a LIFO contractor.

n^{th} arrival is within a busy period. Also, using the triangular inequality, the inequality in (19) implies that

$$0 < |\tau^d(n) - \tau^d(m)| \leq \tau^a(n) - \tau^a(m) \quad (20)$$

for any two packets (with packet indices $m < n$) in the same busy period. This shows that the *absolute values* of interdeparture times are not larger than the interarrival times for those packets in the same busy period.

Clearly, a FIFO contractor is a special case of an absolute contractor. Another special case is a LIFO contractor defined below.

Definition 12: (LIFO contractor) A LIFO contractor is defined exactly the same as a FIFO contractor in Definition 7 except the condition in (7) is replaced by

$$0 < \tau^d(n-1) - \tau^d(n) \leq \tau^a(n) - \tau^a(n-1), \quad (21)$$

whenever $\tau^a(n) \leq \max_{j < n} \tau^d(j)$.

Note from (21) that the departure order for those packets in the same busy period is reversed. Also, since $\tau^d(n-1) > \tau^d(n)$ and $\tau^a(n) > \tau^a(n-1)$, it follows that the delays for those packets in a busy period is *strictly* decreasing, i.e.,

$$\tau^d(n-1) - \tau^a(n-1) > \tau^d(n) - \tau^a(n). \quad (22)$$

In Figure 7, we show a typical sample path of a LIFO contractor.

B. A construction of an absolute contractor via a concatenation of optical memory cells

Consider the concatenation of optical memory cells in Figure 2.

Theorem 13: Under the self-routing policy (as that in Theorem 8), the concatenation of optical memory cells in Figure 2 is an absolute contractor with maximum delay S_M^g if $d_1 = 1$ and $d_i = \prod_{k=1}^{i-1} r_k$, $i = 2, \dots, M$

Note that under the assumption in Theorem 13

$$\begin{aligned} \sum_{k=1}^i (r_k - 1) \cdot d_k &= \sum_{k=1}^i (r_k - 1) \cdot \prod_{\ell=1}^{k-1} r_\ell \\ &= \prod_{\ell=1}^i r_\ell - 1 = d_{i+1} - 1, \end{aligned} \quad (23)$$

where we use the convention that the product is 1 if its range is null. In view of (23), the condition for an absolute contractor in

Theorem 13 is much stronger than that for a FIFO contractor in Theorem 8. Also, we note that

$$S_M^g = \sum_{i=1}^M (r_i - 1) \cdot d_i = \sum_{i=1}^M (r_i - 1) \cdot \prod_{k=1}^{i-1} r_k = \prod_{i=1}^M r_i - 1.$$

Proof. We will prove this by contradiction. As the self-routing policy is a loop-once policy, collision can only occur at the output link of an optical memory cell in Figure 2. Suppose that there are two packets, indexed by n_1 and n_2 , collide with each other at time t at the output link of the i^{th} optical memory cell. Also, suppose that the n_1^{th} (resp. n_2^{th}) packet arrives at the input link at time t_1 (resp. t_2) with delay x_1 (resp. x_2). Without loss of generality, we assume that $n_1 < n_2$ and thus

$$t_1 < t_2. \quad (24)$$

Since these two packets collide with each other at time t at the output link of the i^{th} optical memory cell, we have

$$t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k = t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k = t. \quad (25)$$

From (25) and the complete decomposition property in Proposition 6(v), it follows that

$$\begin{aligned} & t_2 + x_2 - (t_1 + x_1) \\ &= t_2 + \sum_{k=1}^M I_k^g(x_2) \cdot d_k - t_1 - \sum_{k=1}^M I_k^g(x_1) \cdot d_k \\ &= \sum_{k=i+1}^M I_k^g(x_2) \cdot d_k - \sum_{k=i+1}^M I_k^g(x_1) \cdot d_k \\ &= \sum_{k=i+1}^M I_k^g(x_2) \cdot \prod_{\ell=1}^{k-1} r_\ell - \sum_{k=i+1}^M I_k^g(x_1) \cdot \prod_{\ell=1}^{k-1} r_\ell. \end{aligned} \quad (26)$$

Note that every term in both sums in (26) is an integer multiple of $\prod_{\ell=1}^i r_\ell$. Since $t_2 + x_2 \neq t_1 + x_1$ (from the first inequality in (20)), we then have

$$|t_2 + x_2 - (t_1 + x_1)| \geq \prod_{\ell=1}^i r_\ell. \quad (27)$$

Using the second inequality in (20) yields

$$t_2 - t_1 \geq |t_2 + x_2 - (t_1 + x_1)| \geq \prod_{\ell=1}^i r_\ell. \quad (28)$$

On other hand, we have from (25) that

$$\begin{aligned} t_2 - t_1 &= \sum_{k=1}^i I_k^g(x_1) \cdot d_k - \sum_{k=1}^i I_k^g(x_2) \cdot d_k \\ &\leq \sum_{k=1}^i I_k^g(x_1) \cdot d_k \leq \sum_{k=1}^i (r_k - 1) \cdot \prod_{\ell=1}^{k-1} r_\ell \\ &= \prod_{\ell=1}^i r_\ell - 1. \end{aligned} \quad (29)$$

In view of (28) and (29), we reach a contradiction. ■

Remark 14: As commented in [13], many classical switching theories can be used for constructing various optical queues. We note that the argument used in the proof of Theorem 13 is analogous to that for the self-routing property in banyan networks (see e.g., [21], Section 5.6 and [22], Theorem 3.3.18).

Remark 15: As discussed in Section III-B, we can also use the multistage feedforward construction in Figure 4 for an absolute contractor by choosing $d_i = \prod_{k=1}^{i-1} r_k$ in Figure 4. Such a construction will have the crosstalk-free property, i.e., there is at most one packet arriving at an optical crossbar switch at any time. In addition to the benefit of reducing crosstalks, such a property also allows the crossbar switches in Figure 4 to be made by using banyan types of networks, where there is only one unique routing path from an input to an output.

V. N-TO-1 ABSOLUTE CONTRACTORS

In Section III, we have introduced the constructions of absolute contractors with a single input. In this section, we will extend the results to the constructions of absolute contractors with multiple inputs.

Definition 16: (N-to-1 absolute contractor) An N -to-1 absolute contractor is a network element with N input links and one output link. The N input links are indexed from 1 to N . A packet that arrives at time t and at the i^{th} input link, where $1 \leq i \leq N$, is assigned with a system arrival time $Nt + i - 1$. Packets are then ordered by their system arrival times. Let $\tau^a(n)$, $\tau^s(n)$, and $\tau^d(n)$ be the *arrival* time, the *system arrival* time, and the *departure* time of the n^{th} packet, respectively. Suppose that the departure time of a packet is known upon its arrival. An N -to-1 absolute contractor with the range of delay $[d_1, d_2]$ realizes the set of mappings (or sample paths) that satisfy

$$\tau^a(n) + d_1 \leq \tau^d(n) \leq \tau^a(n) + d_2, \quad \text{for all } n, \quad (30)$$

and the following condition:

$$0 < |\tau^d(n) - \tau^d(n-1)| \leq \tau^s(n) - \tau^s(n-1) \quad (31)$$

whenever $\tau^a(n) \leq \max_{j < n} \tau^d(j)$. In particular, if $d_1 = 0$, then it is called an N -to-1 absolute contractor with maximum delay d_2 .

Clearly, if $N = 1$, then the system arrival time of a packet is simply its arrival time. Thus, an N -to-1 absolute contractor with $N = 1$ is the same as an absolute contractor in Definition 11.

As packets are ordered according to their system arrival times, the condition in (31) implies that the absolute values of the interdeparture times are not larger than the inter system arrival times for those packets in the same busy period, i.e.,

$$0 < |\tau^d(n) - \tau^d(m)| \leq \tau^s(n) - \tau^s(m) \quad (32)$$

for any two packets (with packet indices $m < n$) in the same busy period.

Now we argue that an N -to-1 FIFO multiplexer with buffer B (see e.g., [5], [7]) is in fact a special case of an N -to-1

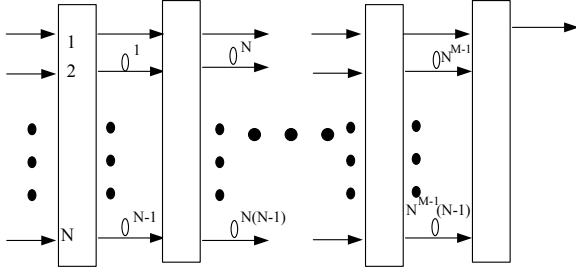


Fig. 8. An N -to-1 absolute contractor.

absolute contractor with maximum delay B . First, as noted in [7], the delay of every packet in an N -to-1 FIFO multiplexer with buffer B is known (by keeping track of the Lindley recursion on the number of packets in the multiplexer). As there is at most one arrival at each input link, we have from the definition of the system arrival time that

$$\tau^s(n) - \tau^s(n-1) \geq 1. \quad (33)$$

Since packets depart contiguously in a busy period in an N -to-1 FIFO multiplexer with buffer B , we have

$$\tau^d(n) - \tau^d(n-1) = 1 \quad (34)$$

for any two consecutive packets in the same busy period. From (33) and (34), it follows that (31) is satisfied for every sample path in an N -to-1 FIFO multiplexer with buffer B . Thus, it can be realized by an N -to-1 absolute contractor with maximum delay B .

In Figure 8, we consider a multistage feedforward construction by optical crossbar switches and fiber delay lines like the one presented in Figure 4. The only differences are that there are N input links in Figure 8 and that the lengths of the delay lines in Figure 8 are chosen from the special case with $r_i = N$ and $d_i = N^{i-1}$, $i = 1, 2, \dots, M$.

To route a packet through the feedforward construction in Figure 8, we also use the generalized \mathcal{C} -transform as in Section III-B. For the special case that $r_i = N$, $i = 1, 2, \dots, M$, the generalized \mathcal{C} -transform of a value x is simply the N -ary representation of x .

Under such a self-routing policy, we note that a packet requesting a delay x will be routed to the delay line with delay jd_i of the i^{th} stage if $I_i^g(x) = j$. Furthermore, if that packet arrives at time t , then that packet will arrive at the output of that delay line at time $t + \sum_{k=1}^i I_k^g(x) \cdot d_k$, where $d_k = N^{k-1}$ for all k .

Theorem 17: Under the above self-routing policy, the multistage feedforward concatenation in Figure 8 is an N -to-1 absolute contractor with maximum delay $N^M - 1$.

Proof. We will prove this by contradiction. Suppose that there are two packets, indexed by n_1 and n_2 , collide with each other at time t at the output of the delay line with delay jd_i of the i^{th} stage (for some i and j). Also, suppose that the n_1^{th} (resp. n_2^{th}) packet arrives at the i_1^{th} (resp. i_2^{th}) input link at time t_1 (resp. the i_2^{th} input link at time t_2) with delay x_1 (resp. x_2). Without loss

of generality, we assume that $n_1 < n_2$. From the definition of the system arrival time in Definition 16, we know that

$$Nt_2 + i_2 - 1 > Nt_1 + i_1 - 1.$$

As both i_1 and i_2 are integers between 1 and N , this then implies that

$$t_2 \geq t_1. \quad (35)$$

Since these two packets collide with each other at time t at the output of the delay line with delay jd_i of the i^{th} stage, we have

$$I_i^g(x_1) = I_i^g(x_2) = j, \quad (36)$$

and

$$t_1 + \sum_{k=1}^i I_k^g(x_1) \cdot d_k = t_2 + \sum_{k=1}^i I_k^g(x_2) \cdot d_k = t. \quad (37)$$

From (37) and the complete decomposition property in Proposition 6(v), it follows that

$$\begin{aligned} & t_2 + x_2 - (t_1 + x_1) \\ &= t_2 + \sum_{k=1}^M I_k^g(x_2) \cdot d_k - t_1 - \sum_{k=1}^M I_k^g(x_1) \cdot d_k \\ &= \sum_{k=i+1}^M I_k^g(x_2) \cdot d_k - \sum_{k=i+1}^M I_k^g(x_1) \cdot d_k \\ &= \sum_{k=i+1}^M I_k^g(x_2) \cdot N^{k-1} - \sum_{k=i+1}^M I_k^g(x_1) \cdot N^{k-1}. \end{aligned} \quad (38)$$

Since we assume that $n_1 < n_2$, we have from the first inequality in (32) that

$$|t_2 + x_2 - (t_1 + x_1)| > 0.$$

As every term in both sums in (38) is an integer multiple of N^i , we then have

$$|t_2 + x_2 - (t_1 + x_1)| \geq N^i. \quad (39)$$

Using the second inequality in (32) yields

$$Nt_2 + i_2 - 1 - (Nt_1 + i_1 - 1) \geq |t_2 + x_2 - (t_1 + x_1)| \geq N^i. \quad (40)$$

On other hand, we have from (37) and (36) that

$$\begin{aligned} t_2 - t_1 &= \sum_{k=1}^i I_k^g(x_1) \cdot d_k - \sum_{k=1}^i I_k^g(x_2) \cdot d_k \\ &= \sum_{k=1}^{i-1} I_k^g(x_1) \cdot d_k - \sum_{k=1}^{i-1} I_k^g(x_2) \cdot d_k \\ &\leq \sum_{k=1}^{i-1} I_k^g(x_1) \cdot d_k \leq \sum_{k=1}^{i-1} (N-1) \cdot N^{k-1} \\ &= N^{i-1} - 1. \end{aligned} \quad (41)$$

Thus,

$$\begin{aligned} Nt_2 + i_2 - 1 - (Nt_1 + i_1 - 1) &= N(t_2 - t_1) + i_2 - i_1 \\ &\leq N(N^{i-1} - 1) + (N-1) = N^i - 1. \end{aligned} \quad (42)$$

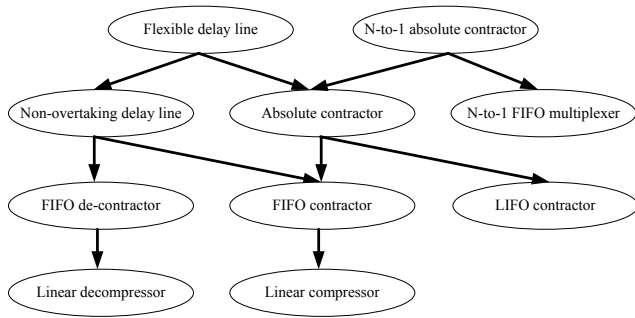


Fig. 9. The relationship among various optical queues.

In view of (42) and (40), we reach a contradiction. ■

VI. CONCLUSIONS

In this paper, we introduced a new class of optical queues that includes FIFO, LIFO and absolute contractors. By the generalized \mathcal{C} -transform, we developed mathematical theories for constructing this new class of optical queues, either by a concatenation of optical memory cells or by a multistage feedforward construction. These new results not only allow us to construct optical queues with switches of arbitrary sizes, but also generalize many known results in the literature, including linear compressors in [13] and the FIFO multiplexers in [5], [7]. In particular, the construction of an absolute contractor in Theorem 13 is the first result that allows overtaking in a concatenation of optical memory cells.

In Figure 9, we show the relationship of this new class of optical queues to various optical queues with known packet delays in the literature. The relation $A \rightarrow B$ in Figure 9 implies that an optical queue A can be used as an optical queue B . For instance, a flexible delay line in [13] can be used as an absolute contractor, which in turn can be used as a FIFO contractor and then a linear compressor in [13].

Due to space limitation, there are some additional results that we are not able to report in detail here.

(i) Feedback constructions of absolute contractors: instead of using multistage feedforward construction discussed in this paper, it is possible to construct an absolute contractor by using a single (large) switch with a set of fiber delay lines connected from its outputs to its inputs. Such an architecture was previously used in [20] for an approximation of a shared buffer switch and in [11], [12] for priority queues.

(ii) Applications of contractors: in [13], one can construct a non-overtaking delay line by a two-stage construction: a linear compressor at the first stage and a linear decompressor at the second stage. Following the same argument, it is possible to construct a LIFO delay line by putting a LIFO contractor at the first stage and a FIFO de-contractor at the second stage. However, in order to specify the delay in each stage, lookahead might be required.

REFERENCES

- [1] R. L. Cruz and J.-T. Tsai, "COD: alternative architectures for high speed packet switching," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 11–21, February 1996.
- [2] D. K. Hunter, D. Cotter, R. B. Ahmad, D. Cornwell, T. H. Gilfedder, P. J. Legg, and I. Andonovic, " 2×2 buffered switch fabrics for traffic routing, merging and shaping in photonic cell networks," *IEEE Journal of Lightwave Technology*, vol. 15, pp. 86–101, January 1997.
- [3] D. K. Hunter, W. D. Cornwell, T. H. Gilfedder, A. Franzen, and I. Andonovic, "SLOB: a switch with large optical buffers for packet switching," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 1725–1736, October 1998.
- [4] E. A. Varvarigos, "The "packing" and the "scheduling packet" switch architectures for almost all-optical lossless networks," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 1757–1767, October 1998.
- [5] C.-S. Chang, D.-S. Lee, and C.-K. Tu, "Recursive construction of FIFO optical multiplexers with switched delay lines," *IEEE Transactions on Information Theory*, vol. 50, pp. 3221–3233, December 2004.
- [6] 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 Transactions on Information Theory*, vol. 52, pp. 4519–4531, October 2006.
- [7] 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," in *Proceedings IEEE International Conference on Computer Communications (INFOCOM'07)*, Anchorage, AK, USA, May 6–12, 2007.
- [8] C.-S. Chang, Y.-T. Chen, and D.-S. Lee, "Constructions of optical FIFO queues," *IEEE Transactions on Information Theory*, vol. 52, pp. 2838–2843, June 2006.
- [9] 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 Transactions on Information Theory*, vol. 53, pp. 1778–1798, May 2007.
- [10] S.-Y. R. Li and X. J. Tan, "Fiber memory," submitted to *IEEE Transactions on Information Theory*, 2006.
- [11] A. D. Sarwate and V. Anantharam, "Exact emulation of a priority queue with a switch and delay lines," *Queueing Systems: Theory and Applications*, vol. 53, pp. 115–125, July 2006.
- [12] H.-C. Chiu, C.-S. Chang, J. Cheng, and D.-S. Lee, "Using a single switch with $O(M)$ inputs/outputs for the construction of an optical priority queue with $O(M^3)$ buffer," in *Proceedings IEEE International Conference on Computer Communications (INFOCOM'07 Minisymposium)*, Anchorage, AK, USA, May 6–12, 2007.
- [13] C.-S. Chang, Y.-T. Chen, J. Cheng, and D.-S. Lee, "Multistage constructions of linear compressors, non-overtaking delay lines, and flexible delay lines," *Proceedings of IEEE INFOCOM 2006*.
- [14] K. Padmanabhan, A. Netravali, "Dilated networks for photonic switching," *IEEE Transaction on Communications*, vol. 35, pp. 1357–1365, 1987.
- [15] W. Kabacinski, "Modified dilated Benes networks for photonic switching," *IEEE Transaction on Communications*, vol. 47, pp. 1253–1259, 1999.
- [16] G. Maier, A. Pattavina, "Design of photonic rearrangeable networks with zero first-order switching-element-crosstalk," *IEEE Transactions on Communications*, vol. 49, pp. 1268–1279, July 2001.
- [17] M. M. Vaez and C.-T. Lea, "Strictly nonblocking directional-coupler-based switching networks under crosstalk constraint," *IEEE Transaction on Communications*, vol. 48, pp. 316–323, February 2000.
- [18] Y. Deng and T. T. Lee, "Crosstalk-Free Conjugate Networks for Optical Multicast Switching," *Journal of Lightwave Technology*, vol. 24, pp. 3635–3645, 2006.
- [19] H.-W. Lan, C.-S. Chang, J. Cheng, and D.-S. Lee, "Constructions and analysis of crosstalk-free optical queues," *High Performance Switching and Routing (HPSR)*, 2008.
- [20] M. J. Karol, "Shared-memory optical packet (ATM) switch," *SPIE vol. 2024 Multigigabit Fiber Communications Systems*, pp. 212–222, 1993.
- [21] J. Hui, *Switching and Traffic Theory for Integrated Broadband Networks*. Boston: Kluwer Academic Publishers, 1990.
- [22] S.-Y. R. Li. *Algebraic Switching Theory and Broadband Applications*. Academic Press, 2001.