Novel MEMS L-Switching Matrix Optical Cross-Connect Architecture: Design and Analysis—Optimal and Staircase-Switching Algorithms

John T. W. Yeow, *Member, IEEE*, and Samer S. Abdallah

***Abstract*—Free-space optical cross connect (OXC) for optical switching has shown promise in replacing traditional electronic switching fabrics. Micromachined optical switches offer superior performance in terms of bit rate and protocol transparency, which make them futureproof; however, they suffer from nonuniformity in port-to-port optical losses, which limit their use in large-scale optical connects. Recent reports on a novel two-dimensional (2-D) OXC architecture, L-switching matrix, presented that it can significantly improve nonuniformity in optical losses among all output ports. The drawback of L-switching matrix is that it is rearrangeably nonblocking (RNB) and not strictly nonblocking. This paper presents solutions to optimize its switching algorithms. A staircase-switching algorithm is proposed to minimize the occurrence of internal blocking conditions.**

***Index Terms*—Free-space optics, microelectromechanical system (MEMS), micromachines, optical cross connects (OXCs), optical switching.**

I. Introduction

THE NEXT generation of all optical networks requires an optical cross connect (OXC) that has the capability to redirect an optical signal from one input port to multiple output ports without an intermediate electronic interface. Microelectromechanical system (MEMS) OXC has shown promise in being the dominant technology in the telecommunication networks. MEMS optical switching fabrics have demonstrated superiority over competing technologies such as bubble-jet switches [1], [2], liquid-crystal switches [3], [4], thermooptical switches [5], [6], and acoustooptical switches [7], [8] in terms of their scalability, insertion loss, polarization-dependent loss (PDL), wavelength dependence, and crosstalk properties.

MEMS switches achieve switching using two approaches: 1) two-dimensional (2-D) MEMS; and 2) three-dimensional (3-D) MEMS. The 2-D MEMS architecture utilizes binary mirrors that can switch "ON" and "OFF." The positioning of the two mirrors achieve the optical switching. Currently, the 2-D MEMS optical switch is based on the conventional crossbar matrix architecture. One of the major drawbacks of the conventional crossbar matrix architecture is that the free-space optical-path difference between the most distance and the least distance paths is significant. The large difference in path lengths results in significant nonuniform insertion losses among output ports [9], [10]. The large nonuniformity in optical losses restricts 2-D MEMS switches to a size of 32×32. The 3-D MEMS switches address the scalability issues of 2-D MEMS switches, but they are costly in terms of stringent optical requirements and the need for complex control electronics [11].

An OXC is characterized by its architecture and switching algorithms. The switch architecture determines the number of pitches that light has to travel in free space. Therefore, a good switch architecture should minimize the free-space propagation distance between the input and output ports. At the same time, the differences in all the port-to-port switching should be minimized as well. The switching algorithm, on the other hand, should optimize the functioning of the switch. It should present minimal and, preferably, no disruption to existing connections when new connections are requested. In previous reports, a novel 2-D MEMS architecture, L-switching matrix, and its switching algorithm are presented [12], [13]. An L-switching matrix switch prototype is demonstrated, and its high yield ratio fabrication process is illustrated elsewhere [14]. Fig. 1(a) shows the scanning electron microscope (SEM) picture of a 4×4 L-switching matrix and Fig. 1(b) shows a single doubly reflective mirror with dimensions $250\ \mu\text{m} \times 500\ \mu\text{m}$ with a thickness of $20\ \mu$m. The mirror is suspended by a pair of $6\ \mu$m-wide torsion bars with a length of $350\ \mu$m and a thickness of $0.4\ \mu$m.

The L-switching matrix has superior optical switching architecture, however, its switching algorithm is only rearrangeably nonblocking (RNB). This paper addresses the drawbacks of the L-switching matrix architecture and its previous switching algorithm. The novel optimal switching and staircase algorithms presented here represent a significant performance improvement over previously proposed ones.

II. L-Switching Matrix: Pros and Cons

Performance issues related to the optimal size of mirrors, spacing between mirrors, loss due to angular mirror or collimator misalignment, and optimal beam-waist size have been addressed [15]. However, one of the most important loss

Manuscript received November 29, 2004; revised July 28, 2005.

J. T. W. Yeow is with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: jyeow@engmail.uwaterloo.ca).

S. Abdallah is with the American University of Beirut, Beirut 1107 2020, Lebanon.

Digital Object Identifier 10.1109/JLT.2005.856292

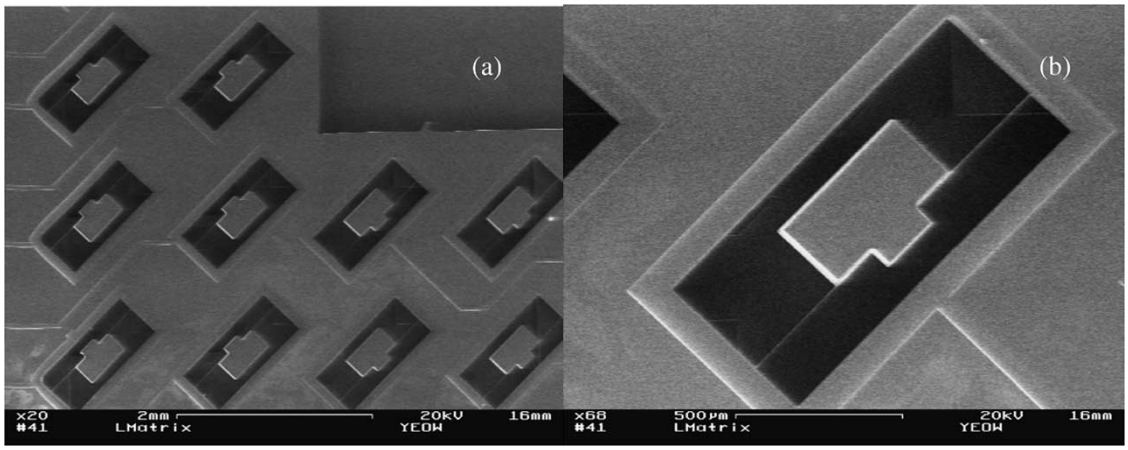

Fig. 1. (a) 4×4 L-switching matrix showing the junction quadrant and the output quadrant I. (b) Close-up view of a doubly reflective mirror.

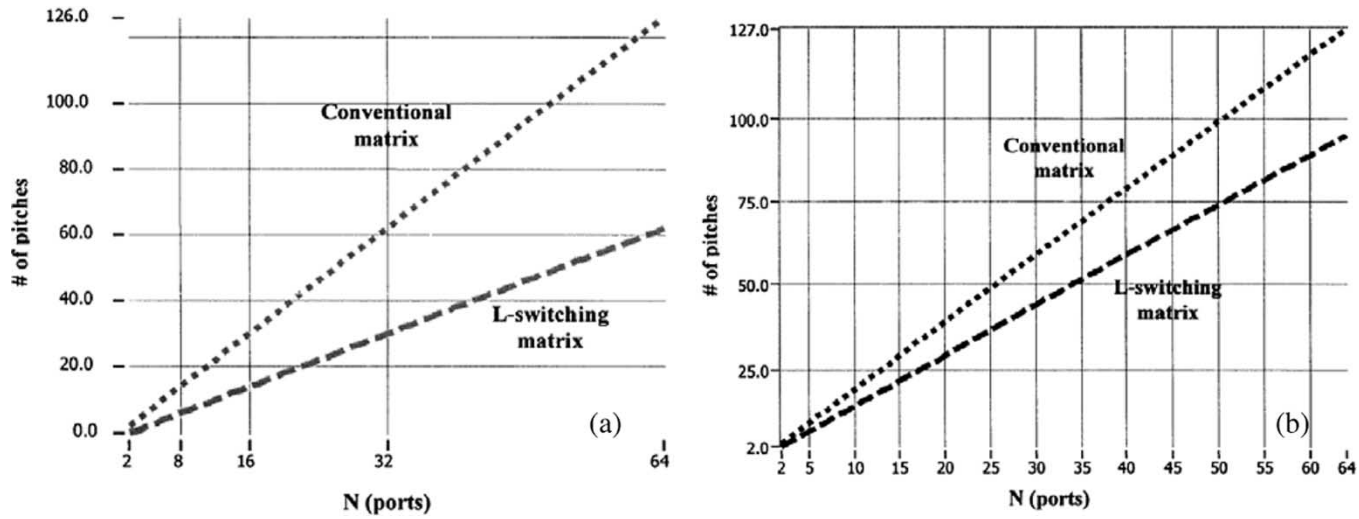

Fig. 2. (a) Most distance paths of conventional crossbar and L-switching matrices. (b) Maximum path-length difference of conventional crossbar and L-switching matrices.

contributors, path-length-dependent loss, has not been addressed. All current free-space optical switches are based on the conventional crossbar architecture. The problems with the conventional crossbar architecture are twofold: 1) beam waist is selected where the insertion loss is the lowest for the most popular path length; therefore, high loss occurs over most and least distance paths, and the loss increases as the number of ports increases; and 2) large nonuniform loss between the best and worst coupling paths because of huge differences in its path lengths. The most and least distance paths of the conventional crossbar architecture are $2N-1$ and 1. On the other hand, the most and least distance paths of the L-switching matrix are $N+((N/2)-1)$ and $((N/2)+1)$. The L-switching matrix lowers the maximum free-space path that light beams travel and hence, lowers the loss due to Gaussian divergence. It is also interesting to note that the least distance path is not constant, as the case for the conventional crossbar matrix, but is dependent on N. Therefore, the L-switching matrix also decreases the maximum difference in interport optical loss. The maximum path differences of the conventional crossbar and L-switching matrices are $2(N-1)$ and $(N-2)$. Fig. 2(a) illustrates the comparison of the maximum distance paths of the conventional crossbar and L-switching matrices. Fig. 2(b) shows the comparison of maximum path differences of interport switching. It can be concluded that the architecture of the L-switching matrix presents an improvement over the conventional crossbar matrix in terms of uniformity in port-to-port and overall optical losses [12]. In addition, the architecture requires smaller die area and less mirrors. Although the L-matrix architecture has superior performance in port-to-port optical losses, it has switching drawbacks that are not found in the conventional crossbar architecture [13]. These switching drawbacks can be minimized with the switching algorithms presented in Sections IV–VI.

For 2-D MEMS optical switches, the following observations are made.

1) The optical switching architecture can be classified into two main categories:
   a) *Without coupling or a decoupled architecture:* The switch architecture is said to be a decoupled architecture if there are independent resources to establish connectivity between any input and any output. The conventional crossbar architecture is clearly a decoupled architecture.

b) *With coupling:* The switch architecture is said to be a coupled architecture if resources are shared for the goal of establishing connectivity between an input and an output. The L-switching matrix clearly has a coupled architecture because it uses double-sided (DS) mirrors and optical paths of some connections might share a pitch in the junction quadrant.

2) Switching algorithms can be classified into two main categories.
   a) *Memoryless algorithm (MLA) (also called Reset-First Algorithm or Always-Rearrangeable Algorithm:* As its name suggests, this kind of algorithm does not take into account past sustained connections. When new pairs of connections are requested, the switch resets itself and establishes new switching configurations by changing the states of the mirrors. The major disadvantage of an MLA is the disruption of existing connections when a new connection pair is requested.
   b) *Memory algorithm (MA):* This algorithm remembers past sustained connections and the state of the pitches (mirrors) that establish these connections. The algorithm tries to establish new connections without disrupting old sustained connections.
3) Given the combination (architecture and algorithm), an MEMS optical switch can be classified into three main categories [16], [17]:
   a) *Strict-sense nonblocking (SSNB) switch:* An optical switch is nonblocking in the strict sense if new connections can always be established without affecting sustained connections in any way. A decoupled architecture (like crossbar architecture) guarantees SSNB property.
   b) *Wide-sense nonblocking (WSNB) switch:* An optical switch is nonblocking in the wide sense if there is an algorithm for establishing paths in the network one after another so that after each path is established, it is still possible to connect any unused input to any unused output.
   c) *RNB switch:* An optical switch is RNB if it is capable of establishing connectivity once all connections are known in advance. Any change in the connection set will require resetting the switch and rerunning the switching algorithm. A switch with MLA is, by definition, RNB.

The L-switching architecture, with its current algorithm, proved to be RNB [13]. It has been shown that the L-switching architecture can support changes in the connection set without resetting the switch in some cases, which means that the penalty of being RNB can be reduced. This interesting property of the L-switching matrix can be maximized by optimizing the RNB algorithm and extending it with the staircase-switching technique. In the rest of paper, the emphasis is on determining the combination of optimal algorithm and staircase-switching technique that reduces the probability of internal blocking.

Section III describes the different types of blocking. Section IV describes the best RNB algorithm to be extended with the staircase-switching technique and compares it to current RNB through illustrative examples. Section V presents a detailed performance analysis in terms of probability of internal blocking. Section VI presents a detailed description of the final algorithm where optimal RNB is integrated with staircase switching. The paper ends with a conclusion that summarizes the relevance of the work presented in this paper.

III. Classification of Switching Conditions

Blocking conditions arise when two connections try to switch their outputs, and, as a result, disrupt existing connections. The blocking conditions constitute a subset in the set of all conditions that occurs when there is an output exchange between two successive connection sets. The coined expression "output exchange" means that two connection pairs in a connection set exchange their outputs in the next connection set. All types of output exchange are classified in the following section and illustrated with a generic example.

A. Types of Blocking Situations

Consider an 8×8 L-switching matrix with the connection set {(1, 5) (2, 2) (3, 3) (4, 8) (5, 4) (6, 7) (7, 6) (8, 1)}. Each pair represents a connection of the form (input, output). Fig. 3 shows the mirror states (MSs) of the switch, given the connection set listed above.

Suppose that any two connection pairs want to perform output exchange. Five mutually exclusive switching conditions will arise.

1) The inputs have same output quadrants:
   a) I_1 and I_4 (they connect to the second output quadrant);
   b) Similarly for: I_2 and I_3; I_5 and I_8; and I_6 and I_7.
2) The inputs are on the same input edge and have different output quadrants with the input doing same-side switching (SSS) closer to the corner of the junction quadrant (closer to the origin) than the one doing cross-side switching (CSS). SSS occurs when the optical beam going into the switch and the optical beam going out of the switch have the same direction. CSS occurs when the optical beam going into the switch and the optical beam going out of the switch have orthogonal directions.
   a) I_2 connects to O_2 in the first quadrant by doing SSS; I_1 connects to O_5 in the second output quadrant by doing CSS; I_2 and I_1 are on the same edge and I_2 is closer to the origin O than I_1;
   b) Similarly for: I_3 and I_1; I_7 and I_8; and I_6 and I_8.
3) The inputs are on the same input edge and have different output quadrants with the input doing CSS closer to the corner of the junction quadrant (closer to the origin) than the one doing SSS:
   a) I_4 connects to O_8 in the second output quadrant by doing CSS; I_2 connects to O_2 in the first quadrant by doing SSS; I_4 and I_2 are on the same edge and I_4 is closer to the origin O than I_2;
   b) Similarly for: I_4 and I_3; I_5 and I_6; and I_5 and I_7.

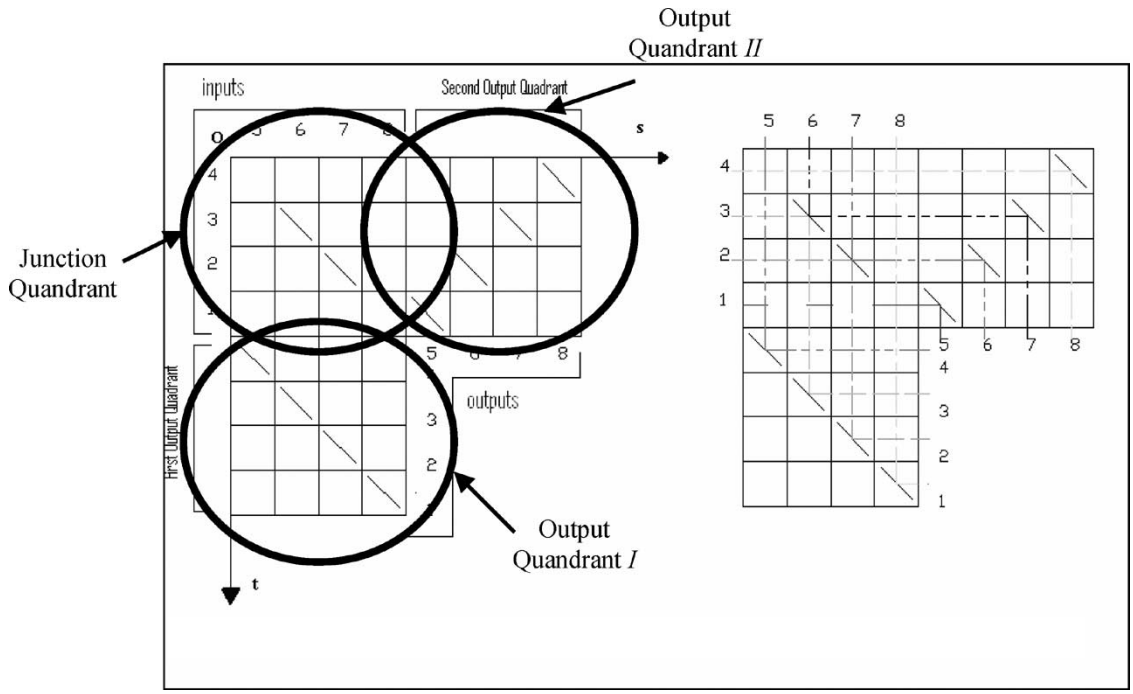

Fig. 3. Status of the L-switching matrix given connection set {(1, 5) (2, 2) (3, 3) (4, 8) (5, 4) (6, 7) (7, 6) (8, 1)}.

4) The inputs are of different input edges and have different output quadrants, but their paths share a cell in the junction quadrant.
   Type 4 is further classified into two subtypes:
   a) 4_c: both inputs are doing CSS:
      i) I_1 and I_5; I_1 and I_8; I_4 and I_5; and I_4 and I_8.
   b) 4_s: both inputs are doing SSS:
      i) I_2 and I_7; and I_3 and I_6.
5) The inputs are of different input edges. They have different output quadrants and do not share any pitch in the junction quadrant. To satisfy the aforementioned criteria, both inputs should be doing SSS for the following reasons.
   a) Given two inputs of different input edges, if one input is doing SSS and the other is doing CSS, they will have the same output quadrant and this contradicts with the definition of type 5 switching.
   b) Given two inputs of different input edges, if both inputs are doing CSS, then they have to share a pitch in the junction quadrant.
      Examples of output switching of type 5 are as follows:
      i) I_2 and I_6; and I_3 and I_7.

Given the previous discussion on the internal blocking, we can derive the mathematical identities for an $N \times N$ switch with C inputs doing CSS and S inputs doing SSS, and with an RNB algorithm.

The number of inputs doing SSS (S) + the number of inputs doing CSS (C) = the total number of input ports (N).

$$S + C = N. \tag{1}$$

Total number of possible output exchange

$$= \binom{N}{2} = \frac{N(N-1)}{2}. \tag{2}$$

Type 1 blocking conditions

$$= 2\binom{\frac{N}{2}}{2} = \frac{N(N-2)}{4}. \tag{3}$$

Type 2 + type 3 blocking conditions

$$= 2\left(\frac{C}{2} \times \frac{S}{2}\right) = \frac{C(N-C)}{2}. \tag{4}$$

Type 4_s + type 5 blocking conditions

$$= \frac{S}{2} \times \frac{S}{2} = \frac{(N-C)^2}{4}. \tag{5}$$

Type 4_c blocking conditions

$$= \frac{C}{2} \times \frac{C}{2} = \frac{C^2}{4}. \tag{6}$$

Total number of possible output exchange

$$= \text{type 1} + \text{type 2} + \text{type 3} + \text{type } 4_s + \text{type } 4_c$$
$$= \frac{N(N-2)}{4} + \frac{C(N-C)}{2} + \frac{(N-C)^2}{4} + \frac{C^2}{4}$$
$$= \frac{N(N-1)}{2}. \tag{7}$$

Equations (2) and (7) should be equal by definition.
It is worth noting that when $C = 0$ (all SSS):

1) Number of type $4_c = 0$
2) Number of type $2 = 0$
3) Number of type $3 = 0$

In order to ensure that the L-switching architecture is strictly nonblocking, solutions to address all five types of output exchange are derived.

1) The first type can be solved because the output quadrants act as a conventional crossbar switch that is nonblocking in the wide and strict sense.
2) In the second type, the initial paths share a pitch in the junction quadrant and thus, turning ON/OFF the mirror in this pitch will achieve output exchange. This technique is given the name of staircase switching.
3) In the third type, the initial paths do not share any pitch in the junction quadrant. Therefore, with the current architecture, there is no algorithmic solution to solve this output-exchange type. The only solution to this problem is to incorporate additional rows and columns in the L-switching matrix to provide redundant resources.
4) This case can be solved by changing the state of the mirror in the shared pitch as in the solution of type 2.
5) Similar to type 3.

IV. Optimal Switching Algorithm

The L-switching matrix consists of a junction quadrant and two output quadrants. The junction quadrant is responsible for directing inputs to their destination output quadrants. The mirrors within the output quadrants are responsible for directing inputs to their destination output ports. Hence, the switching algorithms for the output quadrants are identical and relatively straightforward because the quadrants behave like a conventional crossbar matrix. On the other hand, the algorithm of the junction quadrant is more complex because there are many ways to use an array of DS mirrors for the same connection set. Therefore, the RNB configuration of the L-switching matrix for a particular connection set is not unique.

In the previous section, it has been shown that blocking conditions of types 1, 2, and 4 can be solved by switching algorithms while types 3 and 5 cannot be solved without changing the architecture. Also, (4) shows that the sum of types 2 and 3 is constant, and similarly for type 4_s and type 5 in (5). This means that the optimal RNB algorithm increases the number of blocking situations of types 2 and 4 at the expense of the occurrences of types 3 and 5, respectively. Hence, the cases where the staircase-switching technique could be employed instead of resetting the entire switch are maximized.

The current junction algorithm presented in [13] operates as follows: Starting from the origin, 1) it searches for all inputs doing SSS. From these inputs; 2) it matches a pair that belongs to different edges, and 3) activates the DS mirror shared by the orthogonal paths of the input pair. The current algorithm performs the selection process by considering inputs that are closer to the corner of the junction quadrant (inputs closer to the origin).

A. Example 1: All SSS Scenario

The following example compares the operation of the current algorithm to the optimal algorithm. Given an 8×8 L-switching matrix, with the connection set: {(1, 1) (2, 2) (3, 3) (4, 4) (5, 8) (6, 7) (7, 6) (8, 5)}.

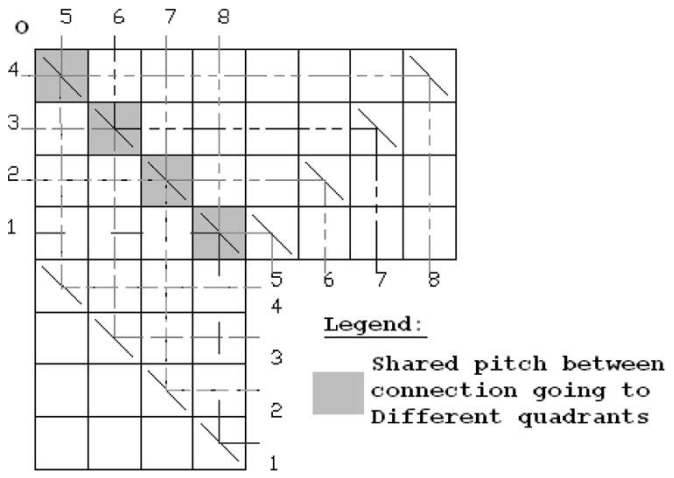

Fig. 4. Status of the L-switching matrix that is running the current algorithm.

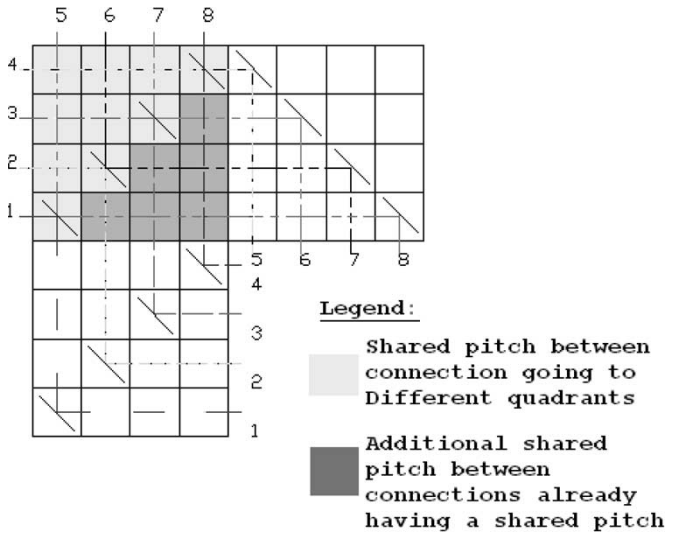

Fig. 5. Status of the L-switching matrix that is running the optimal algorithm.

This situation is a special case because all connections in the connection set are SSS connections.

1) Current Switching Algorithm: Given the above connection set, the current algorithm will produce the switching state presented in Fig. 4.

The resulting type-even (2 or 4) blocking conditions of the switch matrix are: I_4 and I_5; I_3 and I_6; I_2 and I_7; and I_1 and I_8.

This algorithm results in four type-even blocking conditions. At any time, there should be at least $N/2$ type-even blocking conditions in the L-switching matrix. The maximum is $N^2/4$. Therefore, the current algorithm creates the minimum number $(8/2 = 4)$ of type-even blocking conditions.

2) Optimal Switching Algorithm: The optimal algorithm will try to switch DS mirrors, which are common to inputs from the two sides of the input corner, having the largest absolute difference between their indices. The result of using the optimal algorithm is presented in Fig. 5.

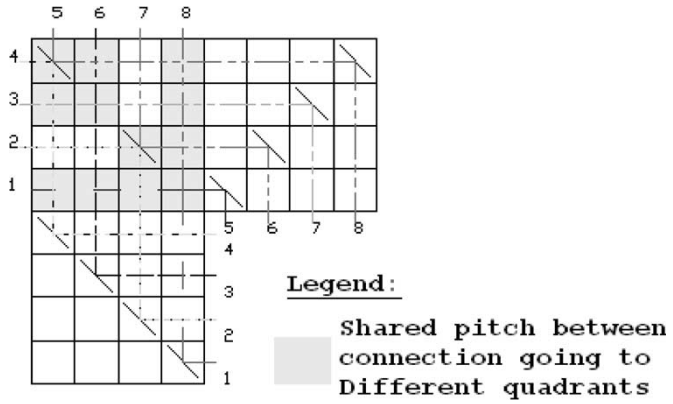

Fig. 6. Status of the L-switching matrix that is running the current algorithm.

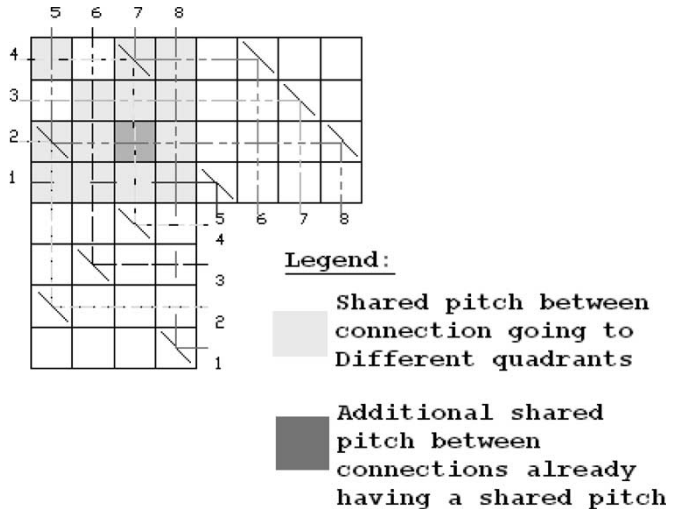

Fig. 7. Status of the L-switching matrix that is running the optimal algorithm.

The resulting type-even (2 or 4) blocking conditions of the switch matrix are: I_4 and I_5; I_4 and I_6; I_4 and I_7; I_4 and I_8; I_3 and I_5; I_3 and I_6; I_3 and I_7; I_2 and I_5; I_2 and I_6; and I_1 and I_5.

The optimal algorithm results in a total of ten blocking type-even conditions that can be solved by the staircase-switching technique. It is worth noting that that, as the number of SSS connections decreases (increasing CSS connections), the difference in the number of type-even blocking conditions between the optimal and the current (worst) algorithms decreases until it disappears when all connections are CSS.

B. Example 2: Mixed Side-Switching Scenario

Suppose we have an 8×8 L-switching matrix, with the connection set {(1, 5) (2, 2) (3, 7) (4, 4) (5, 8) (6, 3) (7, 6) (8, 1)}. The result of using the current algorithm is shown in Fig. 6. The number of type-even blocking conditions is 12. The result of using the optimal algorithm is shown in Fig. 7. The number of type-even blocking conditions is 13. The difference between the two algorithms is decreased to 1. However, both algorithms show a higher number of type-even blocking conditions.

C. Optimal Algorithm: Mathematical Model

The mathematical model of the current algorithm was provided in a previous paper [14]: (8)–(10) constitute the logic equations for implementing the current algorithm.

$$J(s,t) = \bigcup_{m=1}^{\frac{N}{2}} I_{II}^{s}\left(O_{I}^{m}\right) \cap \bigcup_{n=1}^{\frac{N}{2}} I_{I}^{t}\left(O_{II}^{n}\right) \cap \bigcap_{\alpha=1}^{\alpha=s-1} \overline{J(s-\alpha,t)} \cap \bigcap_{\beta=1}^{\beta=t-1} \overline{J(s,t-\beta)} \tag{8}$$

$$O_I(m,t) = I_{I}^{t}\left(O_{I}^{m}\right) \cup \bigcup_{s=1}^{\frac{N}{2}} J(s,t) I_{II}^{s}\left(O_{I}^{m}\right) \tag{9}$$

$$O_{II}(s,n) = I_{II}^{s}\left(O_{II}^{n}\right) \cup \bigcup_{t=1}^{\frac{N}{2}} J(s,t) I_{I}^{t}\left(O_{II}^{n}\right). \tag{10}$$

The optimal algorithm implements a different pattern for SSS connections. It has a similar mathematical model. The optimal algorithm is represented by (11)–(13).

$$J(s,t) = \bigcup_{m=1}^{\frac{N}{2}} I_{II}^{s}\left(O_{I}^{m}\right) \cap \bigcup_{n=1}^{\frac{N}{2}} I_{I}^{t}\left(O_{II}^{n}\right) \cap \bigcap_{\alpha=1}^{\alpha=\frac{N}{2}-s} \overline{J(s+\alpha,t)} \cap \bigcap_{\beta=1}^{\beta=\frac{N}{2}-t} \overline{J(s,t+\beta)} \tag{11}$$

$$O_I(m,t) = I_{I}^{t}\left(O_{I}^{m}\right) \cup \bigcup_{s=1}^{\frac{N}{2}} J(s,t) I_{II}^{s}\left(O_{I}^{m}\right) \tag{12}$$

$$O_{II}(s,n) = I_{II}^{s}\left(O_{II}^{n}\right) \cup \bigcup_{t=1}^{\frac{N}{2}} J(s,t) I_{I}^{t}\left(O_{II}^{n}\right). \tag{13}$$

The mathematical model is a set of Boolean equations. The only difference between the two sets is in the equation of the junction quadrant where they differ in the indices of union and intersection operations.

V. Blocking Analysis

For any state of the L switch, one can compute the number of type-even conditions for the current and the optimal algorithms.

Let C be the number of cross-side connections in the connection set of interest. C is an even number.

Let B be the total number of input pairs (a, b) satisfying the following conditions:

1) "a" is doing CSS;
2) "b" is doing SSS;
3) "a" and "b" belong to the same edge;
4) "a" is closer to the junction corner (origin) than "b;"
5) "a" and "b" do not share any pitches in the junction quadrant.

B can also be considered as the total number of type-3 blocking conditions. For an $N \times N$ switch and for a given C,

any algorithm will give B such that B satisfies the following inequality:

$$0 \le B \le 2 \times \frac{C}{2}\left(\frac{N-C}{2}\right). \tag{14}$$

Proof of (14):

1) $B = 0$ if all inputs doing SSS are closer to the origin than any input doing CSS;
2) $B = B_{\max}$ when all inputs doing CSS are closer to the origin than any input doing SSS.

$$B_{\max} = 2 \times (\#\text{ of inputs doing CSS/edge}) \times (\#\text{ of inputs doing SSS/edge}).$$

It should be noted that B is independent of the algorithm used. B is a function of the connection set only. This fact will be illustrated in the following examples. ■

A. Blocking Analysis of Current Algorithm

For an $N \times N$ L-switch matrix, the number of type $4_{\rm s}$ blocking conditions of the current algorithm:

$$\#\text{ of diagonal pitches of SSS} = \frac{N-C}{2}. \tag{15}$$

Using mathematical identity (5), the number of type-5 conditions with the current algorithm is

$$\begin{aligned} &= \frac{(N-C)^2}{4} - \text{number of type } 4_{\rm s} \\ &= \frac{(N-C)^2}{4} - \frac{(N-C)}{2} \\ &= \frac{(N-C)(N-C-2)}{4}. \end{aligned} \tag{16}$$

Let $E_{\rm c}$ be the number of total type-even blocking conditions. $E_{\rm c}$ consists of two subsets of type-even blocking conditions, $E_{\rm c1}$ and $E_{\rm c2}$.

$$\begin{aligned} E_{\rm c} &= E_{\rm c1} + E_{\rm c2} \\ E_{\rm c1} &= E_{\rm c11} + E_{\rm c12} \\ E_{\rm c11} &= \text{number of type } 4_{\rm c} \text{ blocking conditions} = \left(\frac{C}{2}\right)^2 \\ E_{\rm c12} &= \text{number of type-2 blocking conditions} \end{aligned}$$

and

$$E_{\rm c2} = \text{number of type } 4_{\rm s} \text{ blocking conditions} = \frac{N-C}{2}.$$

From (4) and (14), the number of type 2 + type 3 conditions

$$= \frac{C(N-C)}{2}$$

and the number of type 3 (B)

$$0 \le B \le 2 \times \frac{C}{2}\left(\frac{N-C}{2}\right).$$

Therefore, the number of type 2

$$\begin{aligned} E_{\rm c12} &= \frac{C(N-C)}{2} - B \\ E_{\rm c1} &= E_{\rm c11} + \frac{C(N-C)}{2} - B \\ E_{\rm c} &= E_{\rm c1} + E_{\rm c2} = \frac{CN}{2} - \left(\frac{C}{2}\right)^2 - B + \frac{N-C}{2}. \end{aligned} \tag{17}$$

Let $R_{\rm c}$ be the number of blocking cases of type 3 or 5, and therefore, let $R_{\rm c}$ be the total possible output-switching—number of type even—number of type 1:

$$\begin{aligned} R_{\rm c} &= \binom{N}{2} - E_{\rm c} - 2 \times \binom{\frac{N}{2}}{2} \\ &= \frac{N(N-2)}{4} + \frac{C^2}{4} + \frac{C}{2} + B - \frac{CN}{2}. \end{aligned} \tag{18}$$

When $B_{\max} = C[(N-C)/2]$

$$\text{Max number of type odd} = \frac{N(N-2)}{4} - \frac{C^2}{4} + \frac{C}{2}. \tag{19}$$

B. Blocking Analysis of Optimal Algorithm

The number of type-$4_{\rm s}$ blocking conditions of the optimal algorithm is given by

$$\sum_{a=1}^{\frac{N-C}{2}} a = \frac{\left(\frac{N-C}{2}+1\right)\frac{N-C}{2}}{2} = \frac{(N-C)^2}{8} + \frac{N-C}{4}. \tag{20}$$

Let $E_{\rm o}$ be the number of total type-even blocking conditions in the case of the optimal RNB algorithm.

$$\begin{aligned} E_{\rm o} &= E_{\rm o1} + E_{\rm o2} \\ E_{\rm o1} &= E_{\rm o11} + E_{\rm o12} \\ E_{\rm o11} &= \left(\frac{C}{2}\right)^2 \\ E_{\rm o12} &= \text{number of type-2 blocking conditions} \end{aligned}$$

and

$$E_{\rm o2} = \frac{(N-C)^2}{8} + \frac{N-C}{4}.$$

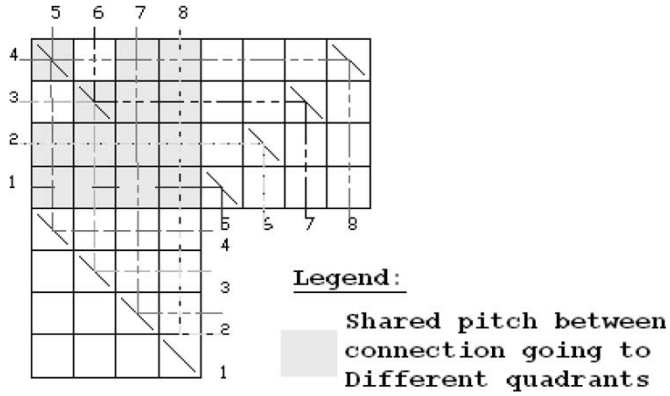

Fig. 8. Status of the L-switching matrix where $C = 4$ and $B = 0$ (current algorithm).

With a similar derivation, the following equations are obtained:

$$E_{\mathrm{o}} = E_{\mathrm{o1}} + E_{\mathrm{o2}} = \frac{CN}{2} - \left(\frac{C}{2}\right)^2 - B + \frac{(N-C)^2}{8} + \frac{N-C}{4}. \quad (21)$$

Let R_{o} be the number of blocking cases of type 3 or 5, and therefore, let R_{o} be the total possible output-switching—number of type even—number of type 1:

$$R_{\mathrm{o}} = \binom{N}{2} - Eo - 2 \times \binom{\frac{N}{2}}{2} = \frac{N(N-2)}{8} + \frac{C^2}{8} + \frac{C}{4} + B - \frac{CN}{4}. \quad (22)$$

When $B_{\max} = C[(N-C)/2]$

Max number of type odd

$$= \frac{N(N-2)}{8} - \frac{3C^2}{8} + \frac{C}{4} + \frac{CN}{4}. \quad (23)$$

1) Illustrations: Fig. 8 shows a state of an 8×8 L switch (using current algorithm) that has four inputs doing SSS $(C = 4)$ (the inputs are: 3, 4, 5, 6) with connection set {(1,5), (2,6), (3,3), (4,4), (5,8), (6,7), (7,2), (8,1)}. Because all these four inputs are closer to the origin (upper left corner of the L switch) than the other inputs (the ones doing same CSS), $B = B_{\min} = 0$. The number of even blocking conditions is given by (17).

Number of type $2 +$ type $4_{\mathrm{c}} +$ type 4_{s}

$$E_{\mathrm{c}} = \frac{CN}{2} - B - \left(\frac{C}{2}\right)^2 + \frac{N-C}{2} = 14.$$

Fig. 9 shows a state of an 8×8 L switch (using current algorithm) that shows four inputs doing SSS $(C = 4)$ (the inputs are: 1, 2, 7, 8) with connection set {(1,1), (2,2), (3,7), (4,8), (5,4), (6,3), (7,6), (8,5)}. Because all these four inputs are farther from the origin (upper left corner of the L switch) than the other inputs, $B = B_{\max} = 8$. The number of even blocking conditions equals 6.

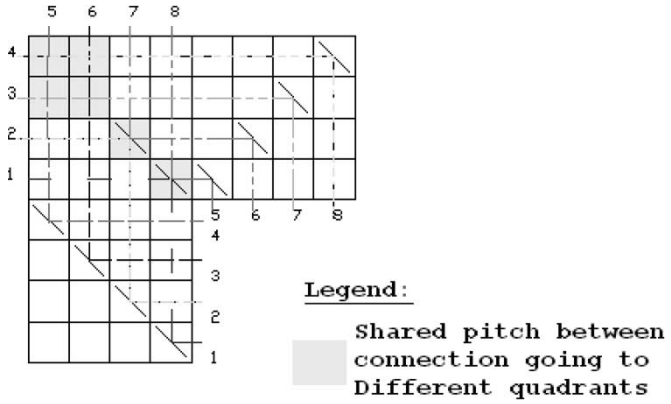

Fig. 9. Status of the L-switching matrix where $C = 4$ and $B = B_{\max} = 8$ (current algorithm).

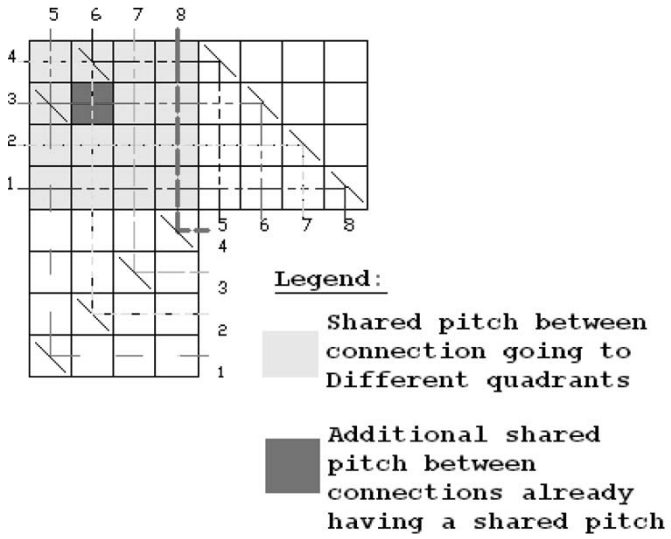

Fig. 10. Status of the L-switching matrix where $C = 4$ and $B = 0$ (optimal algorithm).

Fig. 10 shows a state of an 8×8 L switch (using optimal algorithm) that shows four inputs doing SSS $(C = 4)$ (the inputs are: 3, 4, 5, 6) with connection set {(1,8), (2,7), (3,1), (4,2), (5,6), (6,5), (7,3), (8,4)}. Because all these four inputs are closer to the origin (upper left corner of the L switch) than the other inputs (the ones doing same CSS), $B = B_{\min} = 0$. The number of even blocking conditions is given by (17).

Number of type $2 +$ type $4_{\mathrm{c}} +$ type 4_{s}

$$E_{\mathrm{o}} = \frac{CN}{2} - B - \left(\frac{C}{2}\right)^2 + \frac{(N-C)^2}{8} + \frac{N-C}{4} = 15.$$

Fig. 11 shows a state of an 8×8 L switch (using optimal algorithm) that shows four inputs doing SSS $(C = 4)$ (the inputs are: 1, 2, 7, 8) with connection set {(1,3), (2,4), (3,6), (4,5), (5,1), (6,2), (7,8), (8,7)}. Because all these four inputs are farther from the origin (upper left corner of the L switch) than the other inputs, $B = B_{\max} = 8$. The number of even blocking conditions equals 7.

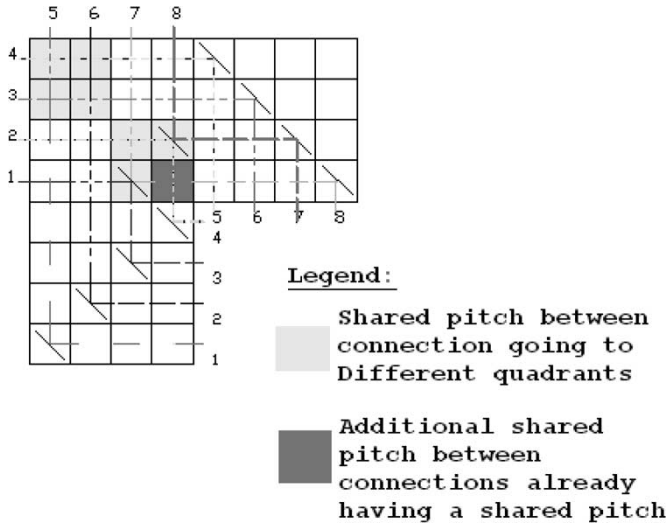

Fig. 11. Status of the L-switching matrix where $C = 4$ and $B = B_{\max}$ (optimal algorithm).

C. Performance Plots

The performance of RNB algorithms (current or optimal) with the staircase-switching technique is determined by computing the probability of needing to reset the switch when faced with the type-3 or type-5 case.

Using (2) and (18), the probability of resetting the switch when using the current algorithm is

$$\Pr(N, C, B) = \frac{\frac{N(N-2)}{4} + \frac{C^2}{4} + \frac{C}{2} + B - \frac{CN}{2}}{\frac{N(N-1)}{2}}. \tag{24}$$

This figure of merit depends on N, the number of CSS connections C (even number), and B, the number of possible output switching of type 3.

Also, using (2) and (23), the probability of resetting the switch when using the optimal algorithm is

$$\Pr(N, C, B) = \frac{\frac{N(N-2)}{8} + \frac{C^2}{8} + \frac{C}{4} + B - \frac{CN}{4}}{\frac{N(N-1)}{2}}. \tag{25}$$

By equating both probabilities for a given N, it is observed that they converge near 0% probability of blocking condition when $C = N - 2$. This is the case where only one pair of inputs is performing SSS and the rest of the inputs are performing CSS. Fig. 12 presents the dependence of the performance of both algorithms on the connection set and, specifically, on the number of cross-side connections C. When C is very small, the performance of the optimal algorithm is 100% better than that of the current algorithm (25% compared to 50% probability of blocking). The performance of both algorithms converges as C increases. The performance of the current algorithm matches that of the optimal algorithm when $C \geq N - 2$. The convergence point is directly proportional to N. At the convergence point, both algorithms show very low probability of blocking. The significance of Fig. 12 is that it shows the sensitivity of the performance of both algorithms to the input/output connection sets. Furthermore, the optimal algorithm consistently shows a lower probability of blocking than the current algorithm. It should be noted that the superiority of the optimal algorithm is more evident as N increases.

Simplification of the expressions of probabilities is required in order to have a figure of merit for comparison. Since B depends on the connection set and it satisfies (14), the mean value in the expressions can be used for simplification of the equation on the probabilities of blocking.

$$\text{Let } B = \overline{B} = \frac{B_{\min} + B_{\max}}{2} = \frac{0 + \frac{C(N-C)}{2}}{2} = \frac{C(N-C)}{4}. \tag{26}$$

Also, the expressions of probabilities can be further simplified if we notice that $C/2$ follows a binomial distribution, because each pair of inputs could have a cross-side connection.

$$\Pr\_C = \text{BinomialPdf}\left(\frac{C}{2}, \frac{N}{2}, 0.5\right)$$

$$\Pr\_C = \binom{\frac{N}{2}}{\frac{C}{2}} \times 0.5^{\frac{N}{2}} \tag{27}$$

$$\Pr(N) = \sum_{\text{all } C} \Pr(N, C) \times \Pr\_C. \tag{28}$$

Equations (27) and (28) show the average probability of blocking as a function of N. Fig. 13 shows a plot of the average performance of both algorithms with and without staircase technique as a function of N. As Fig. 13 shows, the blocking probability when using the optimal algorithm with staircase technique reaches 18.5% asymptotically, while it reaches 25% when using the current algorithm with staircase technique. The 7% difference is significant since it is the percentage of the total number of possible output exchange that is calculated using (2). The asymptotic behavior indicates that the performances of switching algorithms are functions of the architecture and not the size of the switch.

In addition, Fig. 13 also shows that the extension of the RNB algorithms (optimal and current) with staircase switching is very rewarding because the switch does not require resetting for every change in pair of connections, except for approx. 25% of the times when using the current algorithm extended and for approximately 18.5% of the times when using the optimal algorithm. The probability expressions are derived under the assumption that, in general, successive applications of staircase switching do not affect the number of output exchange of even type. This fact makes the analysis accurate for the first change in the connection set, and more or less accurate for the rest of the changes.

VI. Staircase-Switching Technique

In the last section, it has been shown that the optimal algorithm is the best RNB algorithm to be extended with the staircase-switching technique, since it maximizes the number of possible cases where staircase switching can be applied and hence, avoid resetting all connections.

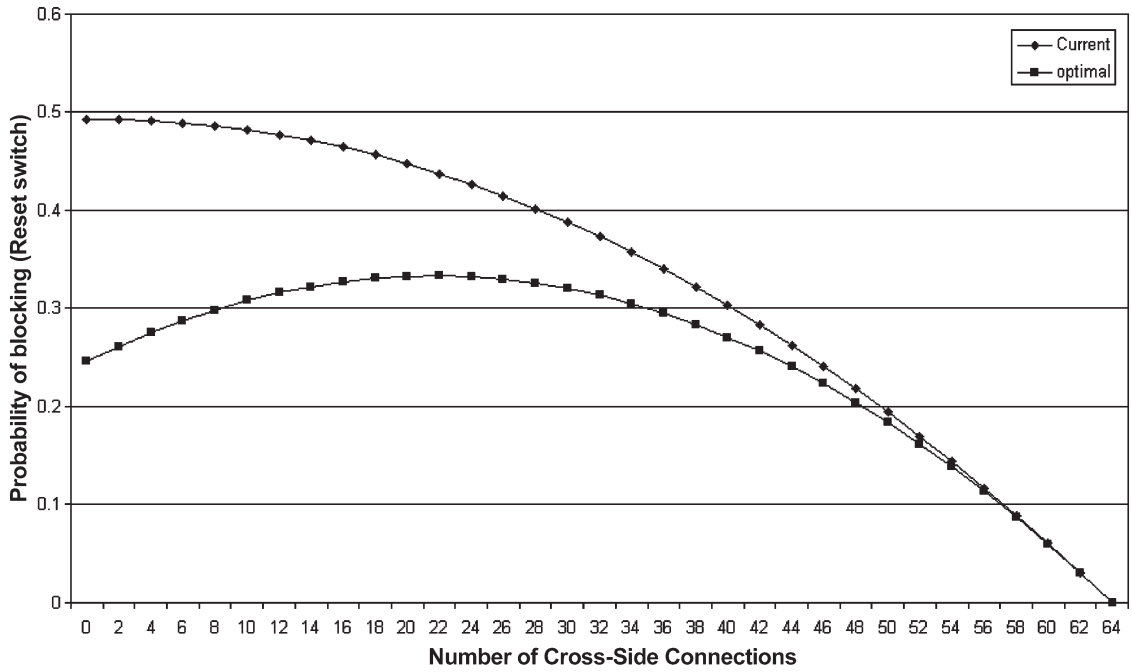

Fig. 12. Performance comparison between current and optimal algorithms for a 64 × 64 L-switching matrix.

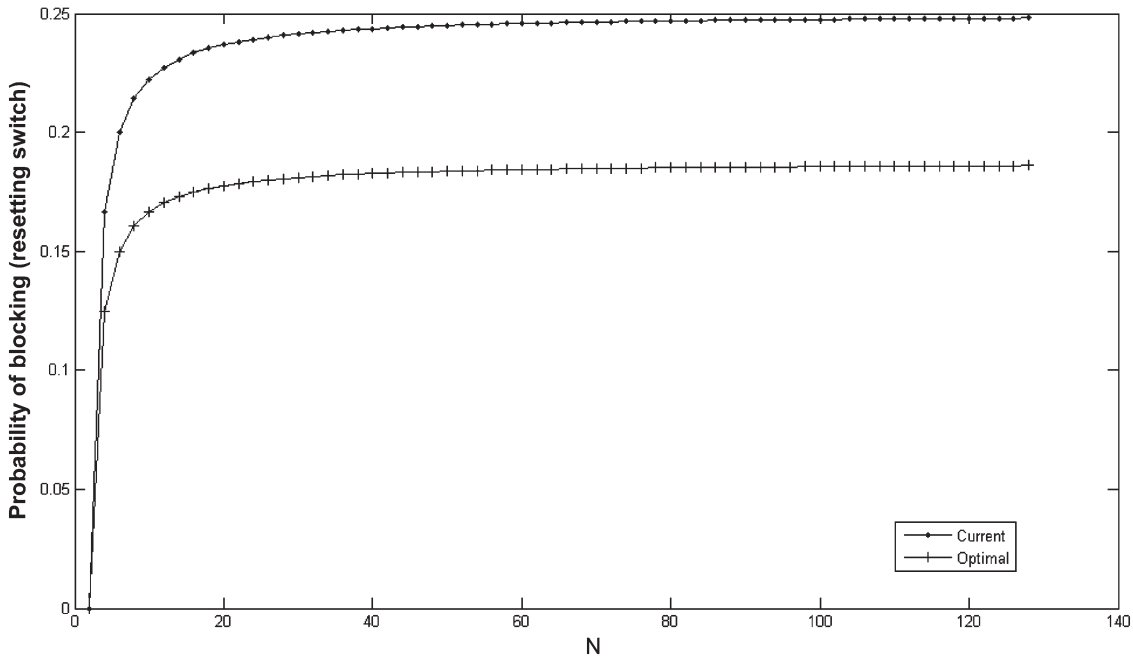

Fig. 13. Performance comparison between current and optimal algorithms, with respect to N, of the L-switching matrix.

In this section, a switching algorithm that combines both the RNB algorithm and the staircase algorithm is presented.

The goal of the staircase-switching technique is to establish a connection set without having to reset the switch. The basic idea of the algorithm presented in this section is to determine the set of mirrors that can be changed in the junction quadrant to establish the new connection set, then to run a switching algorithm over this established set. As presented in the previous sections, there will always be certain conditions where a new connection set cannot be established without having to reset the switch. To implement staircase switching, the MS history of the preceding connection set should be known. The required information will be stored in memory registers.

At all times, the preceding connection set (old connection set) is stored. Fig. 14 shows the coordinate system of the L-switching matrix. The following are the definitions of the connection notations used.

1) $I_b^a(O_d^c) \bullet \text{Old}$ denotes the registers storing the old connection set. $I_b^a(O_d^c) \bullet \text{Old} = 1$ represents that in the previous connection set, input I_b^a was connected to output O_d^c;

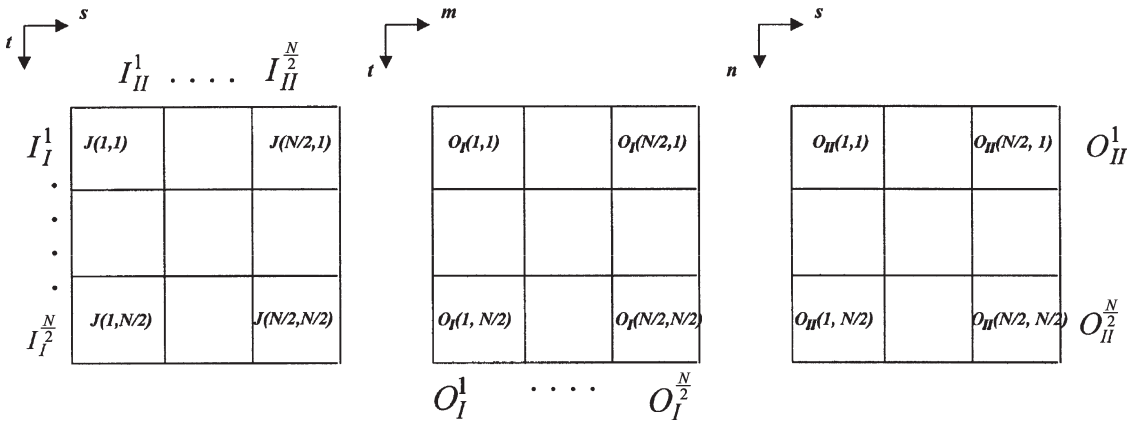

Fig. 14. Coordinate systems for the quadrants in an L-switching matrix.

2) $I_b^a(O_d^c)$ • Current denotes the registers storing the current connection set. $I_b^a(O_d^c)$ • Current $= 1$ represents that in the current connection set, input I_b^a is connected to output O_d^c.

The output quadrant to which each input connects to [**o**ld **o**utput **q**uadrant (**OOQ**)] in the previous connection set is stored and compared to the value of the **c**urrent **o**utput **q**uadrant (**COQ**). From a preceding connection set, the type of output switching, CSS or SSS, of each input is also stored. The old connection set is stored in the **t**ype of **o**ld **c**onnection (**TOC**) register. The current connection set is stored in the **t**ype of **c**urrent **c**onnection (**TCC**) register. The notations I_b^a • TOC and I_b^a • TCC denote the registers for storing the information about old connection type and current connection type, which correspond to input I_b^a, respectively. I_b^a • TOC and I_b^a • TCC are 2-bit registers where:

1) Value stored $= 11$ signifies that the input was doing CSS;
2) Value stored $= 10$ signifies that the input was doing SSS;
3) Value stored $= 0x$ (00 or 01) signifies that the input is not involved in any connection.

The connections that pass through each pitch of the junction quadrant are stored. Given that all inputs are active connections, each pitch will have two connections coming from the "parents" of the pitch. One of the connections is parallel to the vertical axis when it enters the pitch, and the other connection is horizontal.

Any pitch in the junction quadrant has to store the following information:

1) **V**ertical **c**onnection (**VC**) coming from an input—$\lceil \log_2 N \rceil$ bits required to store an input identifier;
2) **H**orizontal **c**onnection (**HC**) coming from an input—$\lceil \log_2 N \rceil$ bits required to store an input identifier;
3) **MS**—1 bit required;
4) **A**vailable **f**or **m**odification **f**lag **b**it (**AFMFB**)—1 bit required.

The dot notation is used to denote each of the registers related to each pitch. Let $J(s,t)$ denote the junction pitch having coordinates (s,t).

1) $J(s,t)$ • VC denotes the VC register related to pitch $J(s,t)$. This register stores the identifier of the input whose connection is entering vertically in the pitch $J(s,t)$.
2) $J(s,t)$ • HC denotes the HC register related to pitch $J(s,t)$. This register stores the identifier of the input whose connection is entering horizontally in the pitch $J(s,t)$.
3) $J(s,t)$ • MS denotes the register where the state of the mirror is saved. A value 1 in this register represents that the mirror in pitch $J(s,t)$ is "ON."
4) $J(s,t)$ • AFMFB denotes the "AFMFB" register. A value of 1 in this register represents that the switching algorithm can change the MS of pitch $J(s,t)$ in order to establish connections.

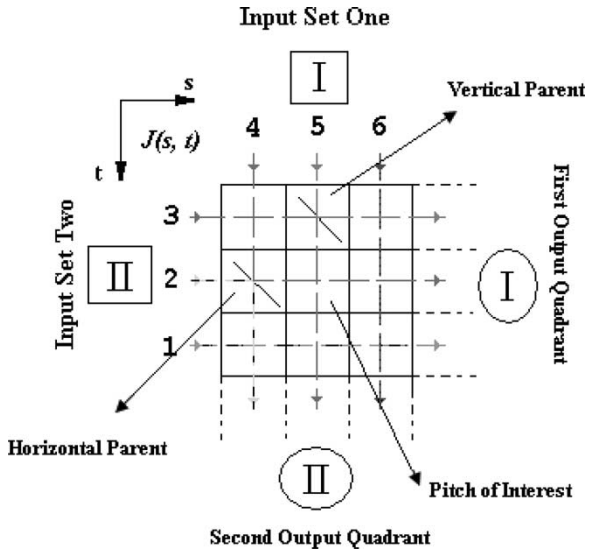

Fig. 15. Portion of junction quadrant showing parents of a pitch of interest.

Fig. 15 shows the junction quadrant for a given connection set. The red and yellow connections pass through the pitch of interest $J(2,2)$. The pitch of interest $J(2,2)$ stores the following information:

1) VC $=$ yellow connection $=$ connection of input $\#$ 3;
2) HC $=$ red connection $=$ connection of input $\#$ 4;
3) MS $=$ "OFF" (Bit $= 0$);
4) AFMFB $=$ "OFF" (Bit $= 0$) (in general).

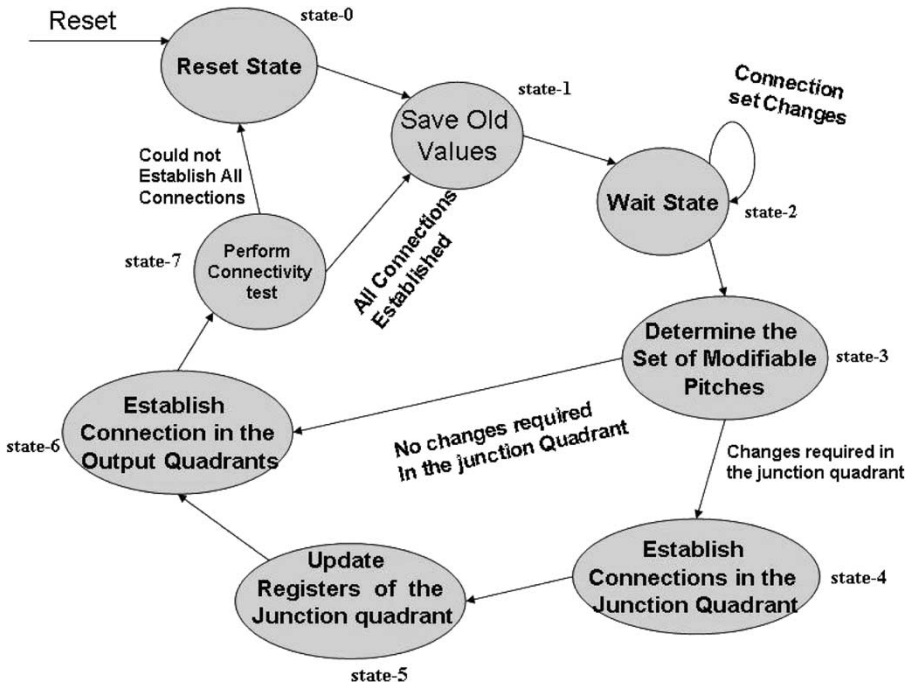

Fig. 16. State diagram for the staircase-switching algorithm.

The staircase-switching technique is best described by the following state diagram shown in Fig. 16. The diagram consists of eight states. In each state, registers are being consulted and/or updated.

The descriptions of each state are provided as follows.

A. State 0: Reset State

In the reset state, all mirrors are switched OFF. The registers belonging to the junction pitches are initialized appropriately as if all connections are assumed to be doing CSS.

The following are the equations used to initialize the registers:

$$\forall a, c \in \left[1, \frac{N}{2}\right] \text{ and } \forall b, d \in \{I, II\}$$

$$I_a^b\left(O_c^d\right) \bullet \text{Current} = \overline{\text{Reset}} \times I_a^b\left(O_c^d\right) \bullet \text{Current} \tag{29}$$

$$I_a^b \bullet \text{TOC}[1] = 1 \tag{30}$$

$$J(s,t) \bullet \text{MS} = 0 \tag{31}$$

$$J(s,t) \bullet \text{VC} = \frac{N}{2} + s \quad \%\text{Set vertical parent} \tag{32}$$

$$J(s,t) \bullet \text{HC} = \frac{N}{2} + 1 - t \quad \%\text{Set horizontal parent.} \tag{33}$$

Equation (29): Equation (29) implements two situations.

1) The first situation occurs if we need to reset everything including the connection set. In this situation, the reset signal will be active $(\text{Reset} = 1)$.

$$I_a^b\left(O_c^d\right) \bullet \text{Current} = 0 \qquad \forall a, b, c, d$$

2) The second situation occurs if the staircase technique fails to establish new connections. As a result, the switch should be reset, but we keep the same connection in order to establish it.

Equation (30): In this equation, the TOC register is reset for every input. This is done by setting the most significant bit (MSB) to 0.

Equation (31): In this equation, the state of each mirror in the junction quadrant is reset to 0 ("OFF").

Equation (32): In (31), all mirrors are set to "OFF." In (32), the VC is set in accordance with the fact that all mirrors are "OFF."

Equation (33): Equation (33) is similar to (32). In (33), the HC is set in accordance with the fact that all mirrors are "OFF."

B. State 1: Save Old Values

In this state, information related to the preceding connection set is saved in the appropriate registers. The following are the equations used in this state:

$$\forall a, c \in \left[1, \frac{N}{2}\right] \text{ and } \forall b, d \in \{I, II\}$$

$$I_b^a\left(O_d^c\right) \bullet \text{Old} = I_b^a\left(O_d^c\right) \bullet \text{Current} \tag{34}$$

$$I_b^a \bullet \text{TOC}[0] = \bigcup_{m=1}^{\frac{N}{2}} I_b^a\left(O_d^m\right) \bullet \text{Old} \tag{35}$$

$$I_b^a \bullet \text{TOC}[1] = \bigcup_{m=1}^{\frac{N}{2}} I_b^a\left(O_d^m\right) \bullet \text{Old} \bigcup_{n=1}^{\frac{N}{2}} I_b^a\left(O_b^n\right) \bullet \text{Old}. \tag{36}$$

Equation (34): Equation (34) saves the old connection set. Note that state 1 can be reached from either state 0 or state 7. If it was from state 0, then what we are saving is a blank connection set. If it was from state 7, then what we are saving

is the old connection set for which we have successfully established all its connections.

Equation (35): In (35) and (36), we save the TOC for each input. Equation (32) saves the least significant bit (LSB) of the TOC register.

Equation (36): In (35), we save the TOC for each input. Equation (36) saves the MSB of the TOC register.

C. State 2: Wait State

In the wait state, we keep monitoring for any changes in the connection set. If the connection set changes, go to state 2. This is the only state where changes to the connection set are permitted.

$$\forall a, c \in \left[1, \frac{N}{2}\right] \text{ and } \forall b, d \in \{I, II\}$$

$$I_b^a \bullet \text{TCC}[0] = \bigcup_{m=1}^{\frac{N}{2}} I_b^a\left(O_d^m\right) \bullet \text{Current} \tag{37}$$

$$I_b^a \bullet \text{TCC}[1] = \bigcup_{m=1}^{\frac{N}{2}} I_b^a\left(O_d^m\right) \bullet \text{Current} \bigcup_{n=1}^{\frac{N}{2}} I_b^a\left(O_b^n\right) \bullet \text{Current} \tag{38}$$

$$\otimes = \text{XOR}$$

$$\text{GotoState2} = \bigcup_{\forall a, c \in \left[1, \frac{N}{2}\right]}^{\forall b, d \in \{I, II\}} \begin{pmatrix} I_a^b\left(O_c^d\right) \bullet \text{Old} \\ \otimes I_a^b\left(O_c^d\right) \bullet \text{Current} \end{pmatrix} \tag{39}$$

$$\text{RedoState1} = \overline{\text{GotoState2}}. \tag{40}$$

Equation (37): In (37) and (38), TCC for each input is saved. Equation (35) saves the LSB of the TCC register.

Equation (38): Equation (38) saves the MSB of the TCC register.

Equation (39): Equation (39) checks if there is a change between the old and the current connection set. In case of change, the next state will be state 2, otherwise, we will keep looping in state 1 until there is a change in the connection set.

Equation (40): Equation (40) determines whether state 1 or state 2 is the next state.

D. State 3: Determine the Set of Modifiable Pitches

This state determines whether the junction quadrant needs to be changed. This situation occurs when an input changes the output quadrant (cross side $\Leftrightarrow$ same side). In the case where the junction quadrant needs to be modified, we determine the set of pitches that are available for modification.

$$\forall a \in \{I, II\} \text{ and } \forall b \in \left[1, \frac{N}{2}\right].$$

Let

$$I_a^b \bullet \text{OQCh} = \left(I_a^b \bullet \text{TOC}[0] \otimes I_a^b \bullet \text{TCC}[0]\right) \bigcup \left(I_a^b \bullet \text{TOC}[1] \otimes I_a^b \bullet \text{TCC}[1]\right) \tag{41}$$

$$J(s,t) \bullet \text{AFMFB} = (J(s,t) \bullet \text{VC}) \bullet \text{OQCh} \bigcap (J(s,t) \bullet \text{HC}) \bullet \text{OQCh} \tag{42}$$

$$\text{GotoState4} = \bigcup_{\forall a \in \{I, II\}}^{\forall b \in \left[1, \frac{N}{2}\right]} I_a^b \bullet \text{OQCh} \tag{43}$$

$$\text{GotoState6} = \overline{\text{GotoState4}}. \tag{44}$$

Equation (41): Equation (41) checks to see if an input changed the output quadrant (junction quadrant should be modified). Also, it checks whether an input is no longer involved in any connection.

Equation (42): Equation (42) determines the pitches in the junction quadrant that can be modified. The condition for having a modifiable pitch is to have both the inputs responsible for the VCs and HCs change their output quadrants.

Equation (43): Equation (43) determines if change the junction quadrants (at least an input changed its output quadrant) is required, or whether the output quadrants need to be modified in order to establish the connection set. If the junction quadrant needs to be changed, signal GotoState4 will be set to 1.

Equation (44): Equation (44) determines whether state 4 or state 6 is the next state.

E. State 4: Establish Connections in the Junction Quadrant

This state implements any RNB algorithm of the L matrix (current and optimal). It has been determined that the RNB algorithm that maximizes the performance of staircase switching is the optimal algorithm that is presented in Section IV. The following are the equations used to determine the mirrors' state in each pitch of the junction quadrant.

Optimal algorithm

$$J(s,t) \bullet \text{MS} = \overline{J(s,t) \bullet \text{MS}} \cap J(s,t) \bullet \text{AFMFB} \bigcap_{\alpha=1}^{\alpha=\frac{N}{2}-s} \overline{J(s+\alpha, t) \bullet \text{AFMFB}} \bigcap_{\beta=1}^{\beta=\frac{N}{2}-t} \overline{J(s, t+\beta) \bullet \text{AFMFB}}. \tag{45}$$

Equation (45): Equation (45) implements the optimal switching algorithm. This equation is similar to (11) but acts only on the set of modifiable mirrors. This equation complements the state of a selected number of pitches (selected

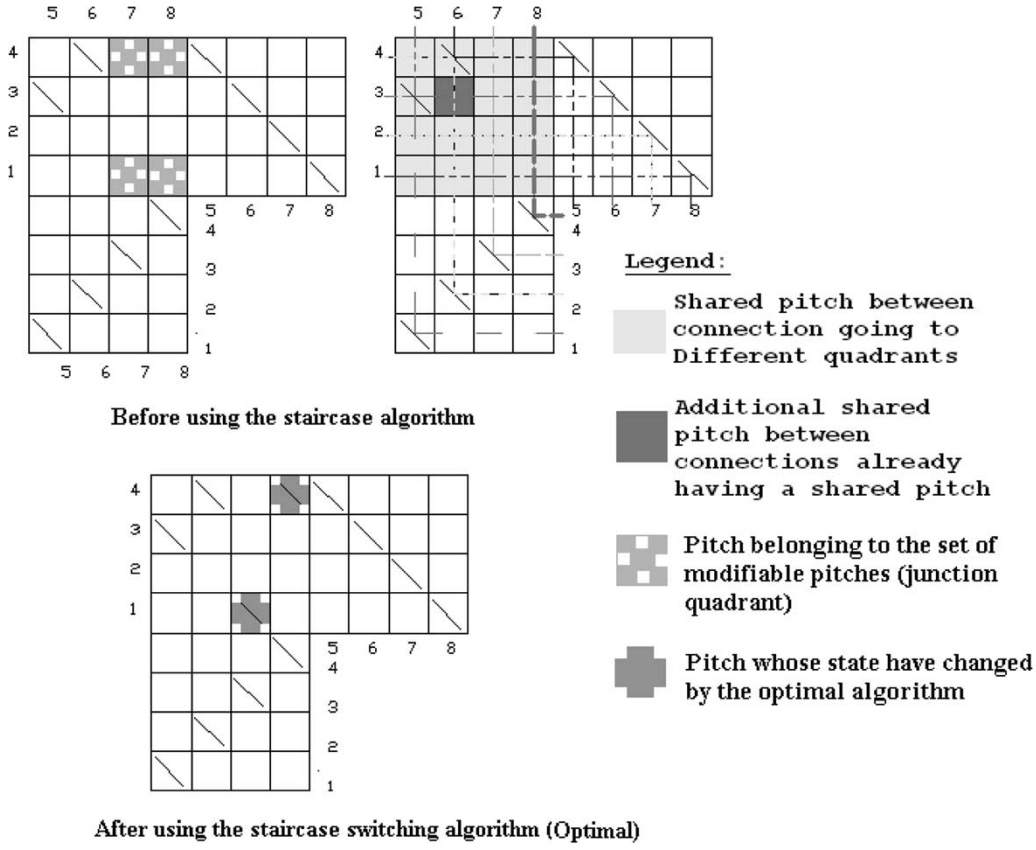

Fig. 17. Example of the operation principles of the optimal and staircase-switching algorithms.

according to the optimal algorithm). Fig. 17 shows the operating principles of the optimal algorithm with staircase-switching technique.

F. State 5: Update Registers of the Junction Quadrant

In the previous state, the state of the mirrors may have changed. This means that we need to update the values of the VC and HC in each pitch.

$$\text{if} \quad (s \neq 1) \text{ and } (t \neq 1)$$

$$J(s,t) \bullet \text{VC} = (J(s,t-1) \bullet \text{HC} \times J(s,t-1) \bullet \text{MS}) + \left(J(s,t-1) \bullet \text{VC} \times \overline{J(s,t-1) \bullet \text{MS}}\right) \tag{46}$$

$$J(s,t) \bullet \text{HC} = (J(s-1,t) \bullet \text{VC} \times J(s-1,t) \bullet \text{MS}) + \left(J(s-1,t) \bullet \text{HC} \times \overline{J(s-1,t) \bullet \text{MS}}\right). \tag{47}$$

Equation (46): Equation (46) sets the VC for each pitch in the junction quadrant.

Equation (47): Equation (47) sets the HC for each pitch in the junction quadrant.

G. State 6: Establish Connections in the Output Quadrants

$O_I(1,t) \bullet$ Input is a register that stores the input identifier connected to the first column of the first output quadrant. This register is used to know which input is entering output quadrant I and from which row.

$O_{II}(s,1) \bullet$ Input is a register that stores the input identifier connected to the first row of the first output quadrant. This register is used to know which input is entering output quadrant II and from which column.

In this state, connections are completely established by modifying output quadrants. The following are the equations used

$$O_I(1,t) \bullet \text{Input} = \left(J\left(\frac{N}{2},t\right) \bullet \text{VC} \times J\left(\frac{N}{2},t\right) \bullet \text{MS}\right) + \left(J\left(\frac{N}{2},t\right) \bullet \text{HC} \times \overline{J\left(\frac{N}{2},t\right) \bullet \text{MS}}\right) \tag{48}$$

$$O_{II}(s,1) \bullet \text{Input} = \left(J\left(s,\frac{N}{2}\right) \bullet \text{HC} \times J\left(s,\frac{N}{2}\right) \bullet \text{MS}\right) + \left(J\left(s,\frac{N}{2}\right) \bullet \text{VC} \times \overline{J\left(s,\frac{N}{2}\right) \bullet \text{MS}}\right) \tag{49}$$

$$O_I(m,t) \bullet \text{MirrorState} = [O_I(1,t) \bullet \text{Input}] \times (O_I^m) \bullet \text{Current} \quad (50)$$

$$O_{II}(s,n) \bullet \text{MirrorState} = [O_{II}(s,1) \bullet \text{Input}] \times (O_{II}^n) \bullet \text{Current}. \quad (51)$$

Equation (48): Equation (48) is used to determine the value stored in $O_I(1,t) \bullet$ Input. The value is either VC or HC of $J[(N/2),t]$ [VC if the mirror in $J[(N/2),t]$ is "ON," HC if the mirror in $J[(N/2),t]$ is "OFF"].

Equation (49): Equation (49) is used to determine the value stored in $O_{II}(s,1) \bullet$ Input. The value is either VC or HC of $J[s,(N/2)]$ [HC if the mirror in $J[s,(N/2)]$ is "ON," VC if the mirror in $J[s,(N/2)]$ is "OFF"].

Equation (50): Equation (50) implements the crossbar switching algorithm in output quadrant I.

Equation (51): Equation (51) implements the crossbar switching algorithm in output quadrant II.

H. State 7: Perform Connectivity Test

Finally, a check is made to determine if the algorithm was able to establish all connections. $I_a^b \bullet$ Connection Established $= I_a^b \bullet$ CE is a 1-bit register that stores the connection status of input I_a^b.

$$\forall t, s$$

$$[O_I(1,t) \bullet \text{Input}] \bullet \text{CE} = \bigcup_{m=1}^{\frac{N}{2}} \{[O_I(1,t) \bullet \text{Input}]\,(O_I^m)\} \bullet \text{Current} \quad (52)$$

$$[O_{II}(s,1) \bullet \text{Input}] \bullet \text{CE} = \bigcup_{n=1}^{\frac{N}{2}} \{[O_{II}(s,1) \bullet \text{Input}]\,(O_{II}^n)\} \bullet \text{Current} \quad (53)$$

$$\text{GotoState1} = \bigcap_{\forall a \in \{I,II\}}^{\forall b \in \left[1,\frac{N}{2}\right]} I_a^b \bullet \text{CE} \cup \overline{I_a^b \bullet \text{TCC}[1]} \quad (54)$$

$$\text{GotoState0} = \overline{\text{GotoState1}}. \quad (55)$$

Equations (52) and (53): Equations (48) and (49) check if the switching algorithm succeeded in establishing the connection of each input in the connection set. In the L-switching matrix, each input will be connected to one output quadrant. All we have to do is to check in each output quadrant whether the inputs connecting to it are required to connect to this quadrant (as in the connection set). Equation (48) does the check in the output quadrant I. Equation (49) does the check in the output quadrant II.

Equations (54): Equation (54) sets the signal GotoState1 to 1 when all connections are established.

Equations (55): If the signal GotoState1 $= 0$ (there is an input whose connection could not be established), the switch is reset, hence, the signal GotoState0 will be set to 1.

VII. Conclusion

In this paper, we have presented an extensive analysis of the L-switching matrix, which determines the maximum performance that can be achieved with the L matrix. Also, we have presented in detail the best switching algorithm for this architecture as a combination of optimal RNB algorithm and staircase-switching technique. The performance analysis done shows a significant reduction in the number of times the switch needs to be reset to implement changes. The analysis done in this paper is useful whenever DS mirrors are employed in future OXC architectures.

References

[1] A. Ware, "New photonic-switching technology for all-optical networks," *Lightwave*, vol. 17, no. 3, pp. 92–98, Mar. 2000.

[2] M. Makihara, M. Sato, F. Shimokawa, and Y. Nishida, "Micromechanical optical switches based on thermocapillary integrated in waveguide substrate," *J. Lightw. Technol.*, vol. 17, no. 1, pp. 14–18, Jan. 1999.

[3] S. Hardy, "Liquid-crystal technology vies for switching applications," *Lightwave*, vol. 16, no. 13, pp. 44–46, Dec. 1999.

[4] N. K. Shankar, J. A. Morris, C. P. Yakymyshyn, and C. R. Pollock, "A 2 × 2 fiber optic switch using chiral liquid crystals," *IEEE Photon. Technol. Lett.* vol. 2, no. 2, pp. 147–149, Feb. 1990.

[5] H. Lausen *et al.*, "Pigtailed thermal optic 1 × 2-switch in polymer: Design and experimental evaluation," in *Proc. Eur. Fiber Optic Conf. Networks*, Heidelberg, Germany, 1994, pp. 99–101.

[6] D. K. Cheng, Y. Liu, and G. J. Sonek, "Optical switch based on thermally-activated dye-doped biomolecular thin films," *IEEE Photon. Technol. Lett.*, vol. 7, no. 4, pp. 366–369, Apr. 1995.

[7] A. Kar-Roy and C. S. Tsai, "8 × 8 symmetric nonblocking integrated acoustooptic space switch module on $LiNbO_3$," *IEEE Photon. Technol. Lett.*, vol. 4, no. 7, pp. 731–734, Jul. 1992.

[8] D. A. Smith, A. Alessandro, J. E. Baran, D. J. Fritz, J. L. Jackel, and R. S. Chakravarthy, "Multiwavelength performance of an apodized acoustooptic switch," *J. Lightw. Technol.*, vol. 14, no. 9, pp. 2044–2051, Sep. 1996.

[9] A. Neukermans and R. Ramaswami, "MEMS technology for optical networking applications," *IEEE Commun. Mag.*, vol. 39, no. 1, pp. 62–69, Jan. 2001.

[10] L. Y. Lin, E. L. Goldstein, and R. W. Tkach, "On the expandability of free-space micromachined optical cross connects," *J. Lightw. Technol.*, vol. 18, no. 4, pp. 482–489, Apr. 2000.

[11] T.-W. Yeow, K. L. E. Law, and A. Goldenberg, "MEMS optical switches," *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 158–163, Nov. 2001.

[12] ——, "Micromachined L-switching matrix," in *IEEE Int. Conf. Communications (ICC)*, New York, May 2002, pp. 2848–2854.

[13] K. L. E. Law, T.-W. Yeow, and A. Goldenberg, "Enhanced designs on MEMS L-switching matrix," in *Proc. IEEE Int. Conf. Communications (ICC)*, Anchorage, AK, May 2003, pp. 1299–1303.

[14] T. W. Yeow, K. L. E. Law, and A. A. Goldenberg, "SOI-based 2D MEMS L-switching matrix for optical networking," *IEEE J. Sel. Topics Quantum Electron.*, vol. 9, no. 2, pp. 603–613, Mar.–Apr. 2003.

[15] L. Y. Lin, E. L. Goldstein, and R. W. Tkach, "Free-space micromachined optical switches for optical networking," *IEEE J. Sel. Topics Quantum Electron.*, vol. 5, no. 1, pp. 4–9, Jan./Feb. 1999.

[16] V. E. Benes, *Mathematical Theory of Connection Networks and Telephone Traffic*. New York: Academic, 1965.

[17] A. Pattavina, *Switching Theory: Architecture and Performance in Broadband ATM Networks*. New York: Wiley, 1998.

John T. W. Yeow (S'99–M'04) received the B.A.Sc. degree in electrical and computer engineering in 1997, and the M.A.Sc. and Ph.D. degrees in mechanical and industrial engineering in 1999 and 2003, respectively, from the University of Toronto, Ontario, Canada.

He is currently an Assistant Professor at the Department of Systems Design Engineering, University of Waterloo, Ontario, Canada. His research interests include the development of microsystem and nanodevices.

Dr. Yeow was the recipient of the 2003 Micralyn Microsystem Design Award and the 2004 NSERC Innovation Challenge Award.

Samer S. Abdallah received the B.Eng. degree in electrical and computer engineering from the American University of Beirut, Beirut, Lebanon, in 2005 and is currently working toward the M.A.S. degree at the Department of Systems Design Engineering, University of Waterloo, Ontario, Canada.