Space and Power Reduction in BDD-based Optical Logic Circuits Exploiting Dual Ports

1st Ryosuke Matsuo
Kyoto University
Kyoto, Japan
matsuo.ryosuke.25r@st.kyoto-u.ac.jp

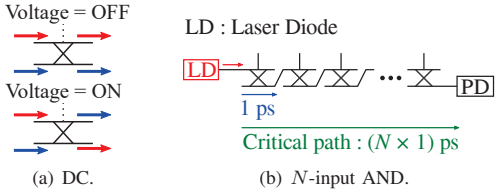
2nd Shin-ichi Minato
Kyoto University
Kyoto, Japan
minato@i.kyoto-u.ac.jp

***Abstract*—Optical logic circuits based on integrated nanophotonics have attracted significant interest due to their ultra-high-speed operation. A synthesis method based on the Binary Decision Diagram (BDD) has been studied, as BDD-based optical logic circuits can take advantage of the speed of light. However, a fundamental disadvantage of BDD-based optical logic circuits is a large number of splitters, which results in large power consumption. In BDD-based circuits a dual port of each logic gate is not used. We propose a method for eliminating a splitter exploiting this dual port. We define a BDD node corresponding to a dual port as a dual port node (DP node) and call the proposed method DP node sharing. We demonstrated that DP node sharing significantly reduces the power consumption and to a lesser extent circuit size without increasing delay. We conducted an experiment involving 10-input logic functions obtained by applying an LUT technology mapper to an ISCSA'85 C7552 benchmark circuit to evaluate our DP node sharing. The experimental results demonstrated that DP node sharing reduces the power consumption by two orders of magnitude of circuit that consume a large amount of power.**

***Index Terms*—Binary Decision Diagram, Optical logic circuit, Logic synthesis**

I. Introduction

Large Scale Integration (LSI) technologies and optical communication technologies have contributed to today's highly advanced information society. Although there has been remarkable progress in LSI technologies since their emergence, the reduction in the total delay per gate saturates at around 10 ps [1]. Optical communication technologies have also been progressing over the past decades. With advances in nanophotonics, optical communication technologies have gradually migrated into ever-shorter distances and moved onto silicon chips as on-chip optical interconnects [2]. Integrated optical circuits using nanophotonic devices have attracted significant interest due to their ultra-high-speed nature. The delay of an optical gate based on a nanophotonic directional coupler (DC) is a few hundred femtoseconds [3], which is more than ten times faster than that of CMOS logic gates.



(a) DC. (b) N-input AND.

Fig. 1. Optical Logic Circuit Using directional coupler (DC).

Many synthesis methods for integrated optical circuits use a DC as a basic building block. A DC has one voltage-control input, two optical inputs, and two optical outputs. The optical routing from inputs to outputs is digitally controlled by the voltage-control signal, as shown in Fig. 1 (a). Like pass transistor logic, Boolean logic can be constructed by serially connecting DCs. Once the input voltage is given, the latency of a DC is determined from only the speed of the light passing through it. In the optical circuit for N-input AND operation, as shown in Fig. 1 (b), the latency is N picoseconds (1 ps delay per DC in this paper). Note that the photodetector (PD) in Fig. 1 (b) receives the light from the laser diode (LD) if and only if all the voltage-inputs are in the ON state. Therefore, the optical circuit in Fig. 1 (b) operates as an N-input AND function. To implement general and larger-scale logic functions, automated design methods that exploit the ultra-high-speed nature of a DC that are based on various schemes have been proposed. Since Binary Decision Diagram (BDD)-based optical logic circuits have ultra-high-speed and area-efficient characteristics, many synthesis methods for optical logic circuits rely on a BDD-based design method. Unfortunately, BDD-based circuits also consume a large amount of power. To address this issue, various design methods for reducing this power consumption have been proposed [4]–[7].

The large power consumption is due to the large number of branches (a branch is called a splitter) in a BDD-based optical logic circuit. Methods for reducing the power loss at a splitter have been proposed [4]–[6]. We propose a method of eliminating splitters by focusing on a node that satisfies a certain condition. We call this node a *dual port node (DP node)* and our method for eliminating a splitter based on DP nodes *DP node sharing*. DP node sharing eliminates splitters by sharing DCs corresponding to DP nodes, which results in a significant power reduction and small reduction in space without increasing delay. We also propose a synthesis method for BDD-based circuits that incorporate DP node sharing. The execution time of the proposed synthesis method is comparable to that of a BDD-based synthesis method that does not incorporate DP node sharing. We conducted an experiment involving 10-input logic functions obtained by applying an LUT technology mapper to an ISCAS'85 C7552 benchmark circuit [8] to evaluate our DP

978-3-9819263-6-1/DATE22/©2022 EDAA

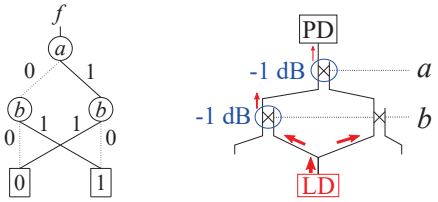
node sharing. The experimental results indicate that DP node sharing exponentially reduces power consumption and slightly reduces circuit size, i.e., number of DCs.

The rest of this paper is organized as follows. In Section II, we describe related work on BDD-based optical logic circuits and issues addressed in this paper. In Section III, we introduce our proposed DP node sharing and its effect on power consumption and circuit size. In Section IV, we introduce our proposed synthesis method for BDD-based optical logic circuits that incorporate DP node sharing. In Section V, we discuss the experimental results. We conclude with a brief summary and mention future work in Section VI.

II. Preliminary–BDD-based Optical Logic Circuits

A. BDD-based Optical Logic Circuits

Fig. 2 (a) shows a BDD representing $f = (\neg a \wedge b) \vee (a \wedge \neg b)$. A solid line represents the 1-edge and a dotted line represents the 0-edge. To build a compact BDD, we check whether a certain node can be removed by using a unique id for each node. Fig. 2 (b) shows the circuit based on the BDD shown in (a). From BDD-based designs [4]–[6], we consider the root and 1-terminal node as the optical output and optical source, respectively. In the optical implementation of a BDD, we use a DC as the BDD node and a splitter as the branch. The electrical inputs corresponding to input variables are fed into DCs corresponding to the BDD nodes to control the direction of the light. The speed of the BDD-based circuit is very fast as determined by the speed of the light passing through serially connected DCs. The computational delay of the circuit for an n-input function is n ps (1 ps delay per DC in this paper). This ultra-high-speed characteristic is the basic motivation underlying the use of BDD-based design for optical logic circuits. The number of BDD nodes is assumed to represent the area of the circuit since the number of DCs is equal to the number of nodes and the area of the splitter is negligible compared with the DC. BDDs can represent general logic functions with a small number of nodes. Therefore, BDD-based optical logic circuits have a high area-efficiency. The fundamental disadvantage of optical circuits is their power consumption. If the intensity of the output optical signal is not sufficiently large, signal-detection time may increase and the circuit may fail to detect due to noise. Unfortunately, input light signals attenuate as they propagate through DCs and splitters. The signal attenuation at DCs is assumed to be -1 dB [9].



(a) BDD. (b) Optical circuit.

Fig. 2. BDD-based Optical Logic Circuit.

When the splitting ratio is $1:1$, a splitter equally splits the power of a single mode input into two single mode outputs, resulting in a signal-power loss of 50% (-3 dB) at one output. Therefore, large power is dissipated in laser sources to maintain a specific signal power level at the output. The power dissipated at an LD is assumed to represent the power consumption of the circuit since the switching energy at DCs is negligibly small. Though we can reduce the signal-power loss at a splitter by optimizing the splitting ratio [5], this method does not resolve the fundamental issue with splitters, i.e., one output of light is always discarded. To address this issue, a method of replacing a splitter with a DC at the cost of increased number of DCs and a delay is proposed [6]. Our DP node sharing can eliminate a splitter while reducing the number of DCs without increasing delay.

B. Power Consumption Model

A method for calculating the optimal splitting ratio and power consumption of an entire circuit is proposed [10]. This method focuses on an insertion loss defined as the total amount of signal power lost on the path from the root to node v. This insertion loss $loss_v$ is formulated by

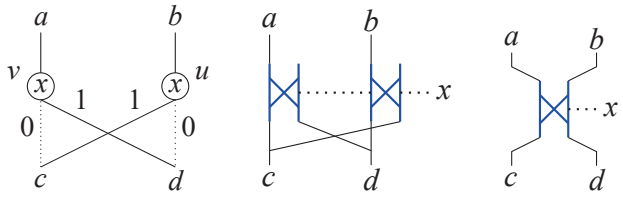
$$loss_v = A \times \sum loss_{parent}. \quad (1)$$

where A is the value for compensating for the signal-power attenuation at DCs. We assume the signal-power attenuation is -1 dB, so we must have $A = 1.25$. The $loss_{parent}$ denotes the insertion loss of the parent nodes of node v. Therefore, Eq. (1) means that $loss_v$ can be obtained by multiplying A by the sum of the insertion loss of parent nodes. However, when the splitting-ratio optimization method [5] is not applied, the insertion loss is larger than $loss_v$. Given the two parent nodes x and y of node v, setting the splitting ratio $SR_x : SR_y$ to $loss_x : loss_y$ minimizes the signal-power loss at the splitter, where $SR_x(SR_y)$ denotes the ratio of signal power split into node $x(y)$. Similarly, given the three parent nodes x, y and z, when $SR_x : SR_y : SR_z = loss_x : loss_y : loss_z$, which minimizes the signal-power loss at the splitter. The same is true when the number of parent nodes is larger than 3. We can calculate the power consumption and optimal splitting ratio of all splitters by calculating Eq. (1) for all nodes from the root to 1-terminal node. Finally, we can obtain the power consumption of the circuit by multiplying a value of the minimum detectable signal power at a PD, which is assumed to be 10 μW, by the insertion loss of 1-terminal node. Therefore, the time complexity is proportional to the number of nodes.

III. DP Node Sharing

A. Concept and Effect

In this section, we describe DP node sharing then discuss its effect on power consumption and number of nodes. Given two nodes v and u labeled with the same variable, we define the two nodes as DP nodes, if the two nodes satisfy the following two conditions: (1) 0-child of v and 1-child of u are the same and (2) 1-child of v and 0-child of u are the same. In Fig. 3 (a), node v and u are a pair of DP nodes. Fig. 3 (b) shows the



(a) DP nodes. (b) Conventional. (c) Share.

Fig. 3. Concept of DP node Sharing.

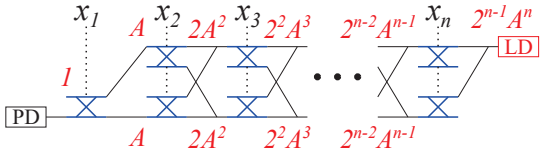


Fig. 4. Conventional BDD-based Circuit.

conventional BDD-based circuit corresponding to the pair of DP nodes v and u. This circuit has two DCs and two splitters. The insertion loss of node c is equal to $A\times(loss_a+loss_b)$ from Eq .1. The same is true for node d. Although it is inevitable that two splitters exist in the part of the circuit corresponding to DP nodes, we can eliminate the two splitters by using a dual port that is not used. Take a look at the left DC. The upper left port and the upper right port are dual. In other words, when the light from node $c(d)$ reaches the upper left port, the light from node $d(c)$ reaches the upper right port. As shown in Fig. 3 (c), we can implement the circuit by a single DC by connecting the right upper port with node b. When $x = 0$, the light from node $c(d)$ reaches node $a(b)$. When $x = 1$, the light from node $c(d)$ reaches node $b(a)$. It is obvious that this operation is logically correct from Fig. 3 (a). In every DC, two output ports are dual as well as two input ports. Therefore, we can use the circuit shown in Fig. 3 (c) for implementation of DP nodes. We define this method as DP node sharing. Note that DP node sharing does not increase the delay of the circuit since there is no extra DC inserted into a path from the LD to PD. Let us consider the insertion loss of node c. The light from node c passes through one DC and reaches node a or b in accordance with the input variable x. No matter which node the light reaches, the signal power has to be larger than the insertion loss of the node that the light reaches. Therefore, the insertion loss of node c is equal to $A \times (\max(loss_a, loss_b))$. The same is true for node d. Since the insertion loss of the circuit with DP node sharing is obviously smaller than that of a conventional BDD-based circuit, DP node sharing can reduce power consumption. Consider the case of $loss_a = loss_b$. In the conventional BDD-based circuit, $loss_c = 2A \times loss_a$. In the circuit incorporating DP node sharing, however, $loss_c = A\times loss_a$. Therefore, the power consumption is reduced by half for every pair of DP nodes, resulting in exponentially smaller power consumption of circuits incorporating DP node sharing compared with that of conventional BDD-based circuits. DP node sharing can also reduce the number of DCs. The number of DCs in Fig. 3 (c) is one less than in Fig. 3 (b), which means that every pair of DP nodes reduces the number of DCs by one. However, the reduction in the number of DCs has a smaller impact than power consumption. Accordingly, DP node sharing is suitable for reducing power consumption rather than the number of DCs. In summary, DP node sharing can reduce both power consumption and number of DCs without increasing delay. Therefore, a circuit is optimized by applying DP node sharing to all DP nodes.

B. Best Case Example

When we implement a function based on a conventional BDD-based optical logic circuit, even if the function can be expressed by a BDD having a small number of nodes, the power consumption varies greatly for each function. Consider examples of AND and XOR. Figs. 1 (b) and 4 show the circuit for AND and XOR, respectively. The number of nodes of AND and XOR is n and $2n-1$, respectively, where n is the number of input variables. Accordingly, the number of DCs of AND and XOR is n and $2n-1$, respectively. Let us consider the power consumption based on an insertion loss. In the circuit for AND, the power consumption is obviously A^n from the calculation method mentioned in Sec II. Consider DCs labeled x_{i+1} and DCs labeled x_i in Fig 4. Since there is a splitter between the DCs, $loss_{x_{i+1}} = A\times loss_{x_i} + A\times loss_{x_i} = 2A\times loss_{x_i}\,(i > 1)$. We must obtain the power consumption of $A^n \times 2^{n-1}$ by calculating an insertion loss from the root node to 1-terminal node. In the circuit for XOR, the increase in the number of DCs is linear, however, that in power consumption is exponential. From the above discussion, the power consumption of the circuit for XOR is exponentially larger than that of AND. However, DP node sharing can drastically reduce the power consumption of XOR. Fig. 5 shows the circuit for XOR incorporating DP node sharing. Consider DCs labeled x_{i+1} and DCs labeled x_i. Since there is no splitter between the DCs, $loss_{x_{i+1}} = A \times loss_{x_i}$, which results in smaller power consumption of A^n. The number of DCs in an entire circuit is n since the number of DCs labeled x_i is 1. In other words, DP node sharing exponentially reduces the power consumption of XOR to that of AND while reducing the number of DCs without increasing delay. In summary, logic functions with huge power consumption, such as XOR, are barriers to practical use of BDD-based optical logic circuits. However, when splitters cause huge power consumption, DP node sharing can sometimes drastically reduce power consumption, which resolves the issue regarding the practicality of BDD-based optical logic circuits.

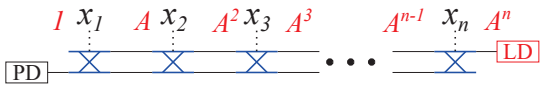


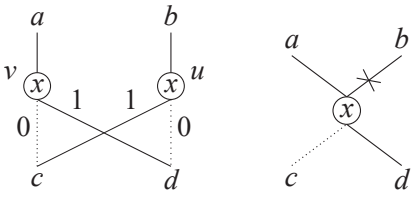
Fig. 5. BDD-based Cicuit Incorporating DP Node Sharing.

IV. Logic Synthesis

In this section, we introduce our proposed synthesis method for BDD-based optical logic circuits incorporating DP node sharing. The flow of our synthesis method is as follows. First, we transform a target logic function into a BDD then design the circuit on the basis of this BDD. When designing the circuit, our synthesis method optimizes connections between DCs and the splitting ratios of splitters.

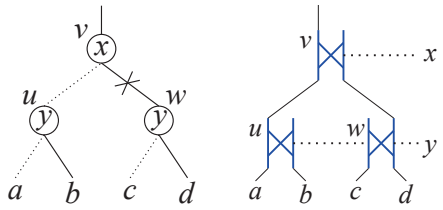
A. BDD Optimization

A BDD incorporating *input inverters* [11] is suitable for the proposed synthesis method. An input inverter is one of the attribute edges, which has a different operation from a complemented edge. Fig. 6 (b) shows a BDD incorporating input inverters corresponding to a BDD shown in Fig. 6 (a). Input inverters indicate an operation of swapping a 0-edge and 1-edge at the next node. This operation is regarded as complementing an input variable. The abuse of input inverters breaks a property that each subgraph uniquely represents a function. However, the uniqueness is maintained under the constraint that the id of a 0-child is smaller than that of a 1-child. Due to this constraint, even if a node does not have a DP node, sometimes that node is connected to its parent node with an input inverter. In a standard BDD, we check whether a certain node can be removed by using a unique id for each node. Similarly, we can check whether a certain node can be connected to its parent node with an input inverter by examining the id of its child nodes. Therefore, implementation of input inverters does not incur additional computational cost compared with the standard BDD. There is a one-to-one correspondence between DP nodes and input inverters, which is clear from the definitions of DP nodes and input inverters as well as Figs. 6 and 3 (a). In a BDD fully incorporating input inverters, every pair of DP nodes is merged into one node. As explained in Section III, circuits incorporating DP node sharing are optimized by applying DP node sharing to all DP nodes. Therefore, the proposed synthesis method applies input inverters to a BDD for the target logic function to fully exploit DP node sharing. Applying input inverters to a BDD makes it is easy to calculate the number of DCs from a BDD since the number of nodes in a BDD incorporating input inverters is equal to the number of DCs in a circuit incorporating DP node sharing.



(a) BDD. (b) Input inv.

Fig. 6. Concept of Input Inverter.



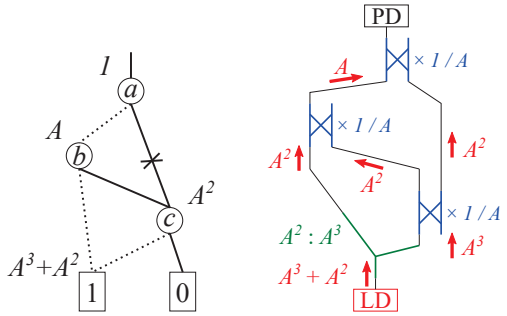
(a) BDD. (b) Circuit.

Fig. 7. Implementation of Input Inverter.

B. Circuit Optimization

We now describe connections between DCs and optimization of splitting ratios. The other factors in circuit design are the same as the conventional BDD-based circuit design explained in Section II. First, we explain how to connect DCs corresponding to nodes connected with an input inverter. Fig. 7 (a) shows a part of a BDD incorporating input inverters. Note that even if a node does not have a DP node, sometimes that node is connected with an input inverter. Consider the operation of input inverters. At a node that is connected to the parent node with a normal edge, when the input variable is equal to 0(1), the 0(1)-child is selected. At a node that is connected to the parent node with an input inverter, when the input variable is equal to 0(1), the 1(0)-child is selected. We use a wiring method to implement this operation. Fig. 7 (b) shows the circuit implementing the BDD shown in Fig. 7 (a). At node $u(w)$, the lower left ports are connected to node $a(c)$ corresponding to the 0-child, and the lower right ports are connected to node $b(d)$ corresponding to the 1-child. We determine which upper port is connected to the parent node depending on whether the node is connected to the parent node with a normal edge or input inverter. Node u is connected to the parent node with a normal edge. In this case, the upper left port is connected to node v so that the light from node $a(b)$ can reach node v when $y = 0(1)$. However, node w is connected with an input inverter. In this case, the upper right port is connected to node v so that the light from node $d(c)$ can reach node v when $y = 0(1)$. We can see that this circuit is logically correct by assigning values to input variables x and y.

Next, we explain a method for calculating an optimal splitting ratio. Section II presented a method for calculating an optimal splitting ratio in a conventional BDD-based circuit based on an insertion loss. However, that calculation method does not incorporate DP node sharing. Therefore, to address DP node sharing, we incorporate the method for calculating the insertion loss at a DP node, explained in Section III, into the calculation method explained in Section II. When we calculate the insertion loss of node v, the correct value can be obtained by just checking the insertion loss of the parent nodes regardless of whether node v is a DP node. Therefore, DP node sharing does not increase the execution time when calculating an optimal splitting ratio. In summary, the execution time of the proposed synthesis method incorporating DP node sharing is comparable to that of a conventional BDD-based synthesis



(a) BDD. (b) Optical circuit.

Fig. 8. Example of Optimal Splitting Ratio with DP Node Sharing.

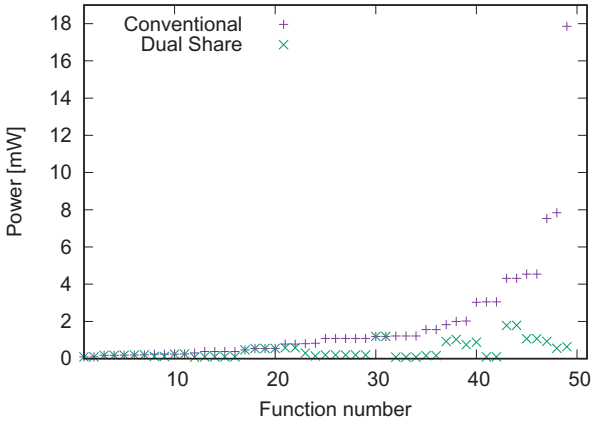


Fig. 9. Results regarding Power Consumption.

method since the execution time of BDD optimization and circuit optimization does not incur additional computational cost. Let us show an example of calculating the optimal splitting ratio in a circuit incorporating DP node sharing. Fig. 8 (a) shows a BDD incorporating input inverters. For sake of simplicity, we define the node labeled by input variable a as node a and the same for nodes b and c. First, $loss_a$ is obviously 1. Second, $loss_b$ is obtained by multiplying A by $loss_a$. Third, since we can apply DP node sharing to node c, $loss_c = A \times max(loss_a + loss_b) = A^2$. Finally, we have $loss_t = A \times (loss_b + loss_c) = A^3 + A^2$, where t denotes 1-terminal node, and $SR_b : SR_c = A^3 : A^2$. Fig. 8 (b) shows the circuit incorporating DP node sharing based on the optimal splitting ratio and power consumption calculated above. Red arrows represent light propagation, blue letters represent signal attenuation at a DC, and green letters represent the splitting ratio. We can see here that sufficient signal power can be obtained at the PD, no matter which path an input light passes through.

V. Experiment and Results

In this section, we discuss the experiment we conducted on how our DP node sharing reduces power consumption and number of nodes. We applied DP node sharing to functions having the same number of inputs to remove the difference in the number of inputs from the experimental results. Therefore, we used the circuits obtained by applying an LUT technology mapper provided by ABC [12] to an ISCAS'85 C7552 benchmark circuit [8]. This circuit-transformation method [7] can reduce the power consumption of large circuits such as ISCAS'85 benchmark circuits by several orders of magnitude. As the number of inputs of an LUT is 10, several logic functions with 1-10 input variables are obtained. We show the results of 10-input logic functions. In our evaluation of the performance of optical logic circuits, we assumed that the power attenuation in a DC is -1 dB [13]. The minimum optical signal-power that the PD can detect is assumed to be 10 μW. We compared the synthesis results of conventional BDD-based circuits and BDD-based circuits incorporating DP node sharing. Since the power consumption of a BDD-based optical circuit largely depends on the variable order, we examine all variable orders to fully exploit DP node sharing.

Fig. 9 shows the results regarding power consumption when the BDDs have variable orders that minimize power consumption. "Conventional" represents the synthesis results of the conventional BDD-based circuits, and "Dual Share" represents those of the BDD-based circuits incorporating DP node sharing. The functions are labeled a number from 1 in ascending order of power consumption of "Conventional". "Function number" represents the number assigned to each function. From the results of "Conventional", we can see that the power consumption varied greatly from function to function in the conventional BDD-based circuits. The smallest power consumption was 93.1 μW, which is the smallest among all feasible 10-input functions. DP node sharing could not reduce the power consumption for this function since there was no splitter. Similarly, DP node sharing can not largely reduce the power consumption when circuits have a small number of splitters. However, conventional BDD-based circuits for such functions consume a small amount of power due to a small number of splitters. Consider the power consumption of function numbers 1-22. DP node sharing did not largely reduce the power consumption. However, since the power consumption of the conventional BDD-based circuits was small, these functions are not barriers to practical use of BDD-based optical logic circuits. Note that the primary goal of DP node sharing is to reduce the power consumption of functions for which conventional BDD-based circuit consumes a large amount of power. Consider the power consumption of function number 40-49. DP node sharing did significantly reduce the power consumption, which means that the cause of high power consumption in these functions is splitters corresponding to DP nodes. The highest power consumption was 17.9 mW, which was reduced to 643 μW with DP node sharing. In other words, DP node sharing, which does not require complicated implementation of software or hardware, can drastically reduce the power consumption of a general logic circuit such as

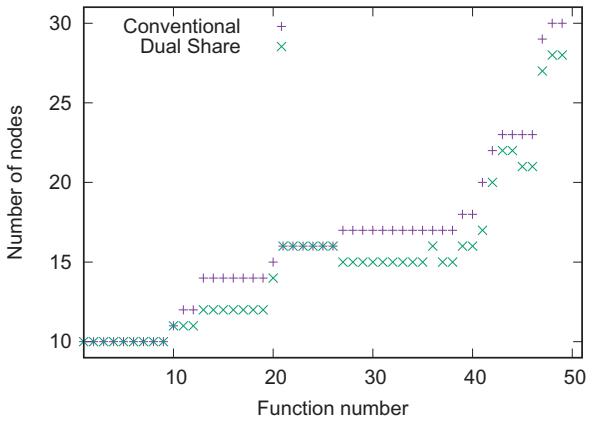


Fig. 10. Results regarding number of Nodes.

ISCAS'85 C7552 benchmark circuit. Let us consider the results regarding the number of nodes. Note that the optimal variable order of "Dual Share" and "Conventional" can differ. Therefore, the number of nodes of "Dual Share" may be larger than that of "Conventional". However, this was the case for just five functions. In four functions, the number of nodes increased by just 1 or 2. In one function, the number of nodes increased by 12. Therefore, we also consider the number of nodes in optimization for variable ordering when a design constraint includes circuit size.

Next, we discuss the results when the BDDs have variable orders that minimize the number of nodes. In all functions, the power consumption of "Dual Share" is not larger than that of "Conventional". Fig. 10 shows the results regarding the number of nodes. The functions are labeled a number from 1 in ascending order of the number of nodes of "Conventional". "Function number" represents the number assigned to each function. We can see that reduction in the number of nodes was much smaller than that of power consumption, which is consistent with the discussion in Section III. Therefore, DP node sharing is suitable for reducing power consumption.

VI. Conclusion

We clarified that circuits having a large number of splitters, such as XOR, consume a large amount of power. To address this issue, we proposed DP node sharing as a method for eliminating splitters and that can be easily implemented in an optical circuit. We also proposed a synthesis method that does not incur additional computational cost compared with a conventional BDD-based synthesis method optimization. Experimental results obtained using ISCAS'85 C7552 benchmark circuit indicate that our DP node sharing drastically reduces the power consumption of optical logic circuits that consume a large amount of power. In other words, DP node sharing is key to breaking down the barrier to practical applications of BDD-based optical logic circuits. Our future work will be focused on developing a logic synthesis method for more reducing the number of splitters in an optical logic circuit.

Acknowledgment

This work is partly supported by JST-CREST JPMJCR18K3 and JSPS KAKENHI Grant Numbers JP21J12765.

References

[1] A. Ceyhan, M. Jung, S. Panth, S. K. Lim, and A. Naeemi, "Impact of Size Effects in Local Interconnects for Future Technology Nodes: A Study Based on Full-Chip Layouts," in *Proc. IEEE Interconnect Technology Conference/Advanced Metallization Conference*, May 2014, pp. 345–348.

[2] Y. Vlasov, "Silicon Photonics for Next Generation Computing Systems," in *Proc. European Conference on Optical Communications*, no. Tu.1.A.1, Sept. 2008.

[3] N. Yamamoto, T. Ogawa, and K. Komori, "Photonic Crystal Directional Coupler Switch with Small Switching Length and Wide Bandwidth," *Optics Express*, vol. 14, no. 3, pp. 1223–1229, Feb. 2006.

[4] R. Wille, O. Keszocze, C. Hopfmuller, and R. Drechsler, "Reverse BDD-based Synthesis for Splitter-Free Optical Circuits," in *Proc. Asia-South Pacific Design Automation Conference*, Jan. 2015, pp. 172–177.

[5] Z. Zhao, Z. Wang, Z. Ying, S. Dhar, R. T. Chen, and D. Z. Pan, "Logic Synthesis for Energy-Efficient Photonic Integrated Circuits," in *Proc. Asia-South Pacific Design Automation Conference*, Jan. 2018.

[6] R. Matsuo, J. Shiomi, T. Ishihara, H. Onodera, A. Shinya, and M. Notomi, "BDD-based Synthesis of Optical Logic Circuits Exploiting Wavelength Division Multiplexing," in *Proc. Asia-South Pacific Design Automation Conference*, Jan. 2019, pp. 355–360.

[7] R. Matsuo, J. Shiomi, T. Ishihara, H. Onodera, A. Shinya, and M. Notomi, "A synthesis method for power-efficient integrated optical logic circuits towards light speed processing," in *2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2020, pp. 488–493.

[8] B. F. and F. H., "A neural netlist of ten combinational benchmark circuits and translator in Fortran," in *Int. Symp. Circuits and Systems (ISCAS)*, June 1985, pp. 663–698.

[9] H. C. Nguyen, N. Yazawa, S. Hashimoto, S. Otsuka, and T. Baba, "Sub-100 μm Photonic Crystal Si Optical Modulators: Spectral, Athermal, and High-Speed Performance," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 19, no. 6, pp. 127–137, June 2013.

[10] R. Matsuo, S. Minato, "BDD Variable Ordering for Minimizing Power Consumption of Optical Logic Circuits," in *2021 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2021, pp. 96–101.

[11] S. Minato, N. Ishiura, and S. Yajima, "Shared Binary Decision Diagram with Attributed Edges for Efficient Boolean Function Manipulation," in *Proceedings of the 27th ACM/IEEE Design Automation Conference*, ser. DAC '90, New York, NY, USA, 1991, pp. 52–57.

[12] Robert Brayton and Alan Mishchenko, "ABC: An Academic Industrial-Strength Verification Tool," in *Proc. of International Conference on Computer Aided Verification*, 2010, pp. 24–40.

[13] N. Yamamoto, T. Ogawa, and K. Komori, "Photonic crystal directional coupler switch with small switching length and wide bandwidth," *Optics Express 14*, Feb 2006.