Journal of Engineering Science and Technology Review 11 (1) (2018) 28- 34

Research Article

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

Verification Method for Area Optimization of Mixed-Polarity Reed-Muller Logic Circuits

Chuandong Chen[1], Bing Lin[2,*] and Michelle Zhu[3]

[1]*Department of microelectronics, Fuzhou University, Fuzhou 350116, China*
[2]*College of Physics and Energy, Fujian Normal University, Fuzhou 350117, China*
[3]*Computer Science Department, Montclair State University, Montclair, NJ 07043, United States*

Received 22 July 2017; Accepted 18 January 2018

Abstract

Area minimization of mixed-polarity Reed-Muller (MPRM) logic circuits is an important step in logic synthesis. While previous studies are mainly based on various artificial intelligence algorithms and not comparable with the results from the mainstream electronics design automation (EDA) tool. Furthermore, it is hard to verify the superiority of intelligence algorithms to the EDA tool on area optimization. To address these problems, a multi-step novel verification method was proposed. First, a hybrid simulated annealing (SA) and discrete particle swarm optimization (DPSO) approach (SADPSO) was applied to optimize the area of the MPRM logic circuit. Second, a Design Compiler (DC) algorithm was used to optimize the area of the same MPRM logic circuit under certain settings and constraints. Finally, the area optimization results of the two algorithms were compared based on MCNC benchmark circuits. Results demonstrate that the SADPSO algorithm outperforms the DC algorithm in the area optimization for MPRM logic circuits. The SADPSO algorithm saves approximately 9.1% equivalent logic gates compared with the DC algorithm. Our proposed verification method illustrates the efficacy of the intelligence algorithm in area optimization compared with DC algorithm. Conclusions in this study provide guidance for the improvement of EDA tools in relation to the area optimization of combinational logic circuits.

Keywords: Area optimization, Intelligence algorithm, Logic synthesis, Mixed-polarity Reed-Muller

1. Introduction

Area optimization of combinational logic circuit is a key step in logic synthesis. Fig. 1 shows the pipelining design of a digital integrated circuit [1]. The combinational logic circuit between triggers is represented by ellipses. This study focused on area optimization of the combinational logic circuit between two triggers. Moreover, this study focuses on the combinational logic circuit with a Reed-Muller (RM) logic circuit [2,3]. The mixed-polarity RM (MPRM) logic circuit is a standard form of the RM logic circuit [4,5]. Area optimization presents a crucial problem for the MPRM logic circuit [6,7] and is the key step in logic synthesis in RM circuits [8].

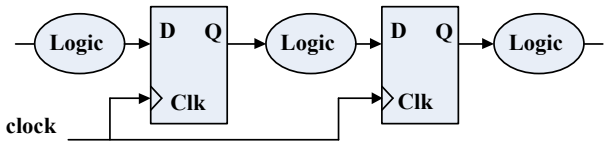

Fig. 1. Pipelining design of digital integrated circuit

Area optimization of the MPRM logic circuit is a nondeterministic polynomial time (NP) complete problem [9,10]. It involves an MPRM combinational logic circuit with n-input m-output and has 3^n mixed polarities. Current studies primarily utilized intelligence algorithms for area minimization of MPRM logic circuits and searched the connection mode of combinational logic circuit with the minimum area through algorithm optimization while ensuring consistent logic functions. Area and timing optimization are key problems in logic synthesis. The Design Compiler (DC) developed by Synopsys® is a relatively authoritative logic synthesis tool. DC is integrated with the area optimization algorithm for logic circuit and can effectively optimize an area of the logic circuit through a standard cell library and logic synthesis constraint.

Abundant experimental data and conclusions are available regarding the area optimization of an MPRM logic circuit based on intelligence algorithms. However, the experimental results of intelligence algorithms cannot be compared with those of mainstream electronics design automation (EDA) tools, thus making the verification of the superiority of intelligence algorithms over EDA tools impossible.

To address these problems, this study proposes a novel related verification method. The proposed verification method enables contrast analysis between the experimental results of the intelligence algorithm and the optimization outcomes of the EDA tool in order to evaluate the optimization ability of the intelligence algorithm.

2. State of the art

MPRM is a standard form of a logic circuit. The area optimization of the MPRM logic circuit, the key step in the area optimization of a logic circuit, has been explored in

*E-mail address: cdchen@fzu.edu.cn
ISSN: 1791-2377
doi:10.25103/jestr.111.04

numerous studies. Genetic algorithm (GA) [11], simulated annealing (SA) [12,13], and discrete particle swarm optimization (DPSO) [14,15] are the major algorithms applied for the area optimization of MPRM logic circuit.

On the basis of research on discrete ternary particle swarm optimization, Hai-Zhen [16] proposed a ternary diversity particle swarm optimization. A mathematic mode for area and low power dissipation was built and mixed polarity conversion of XNOR/OR circuits was improved. The algorithm was tested by using MCNC benchmark circuits. Experimental results showed that the algorithm significantly outperformed other similar methods. However, the experimental results were not compared with the optimization outcomes of the EDA tool. Thus, determining whether this algorithm is better than the EDA tool in terms of area optimization is impossible.

To improve the efficiency of the polarity optimization of MPRM logic circuits, He [17] proposed an efficient and fast polarity optimization approach (FPOA) considering the polarity conversion sequence. FPOA performed better for a complicated MPRM logic circuit. Nevertheless, experimental data on this algorithm were not compared with optimization outcomes of the EDA tool.

Zhang [18] proposed an innovative niche GA for area optimization of fixed-polarity RM circuits. Experimental results of the MCNC benchmark circuits showed that the proposed algorithm was superior to the traditional GA. However, this study compared niche GA and traditional GA but did not compare the optimization results of the niche GA with those of the EDA tool.

Wang [19] presented a power estimation model for MPRM logic circuits, which accurately and efficiently handled temporal signal correlations during the estimation of average power by using lag-one Markov chains. An ordered binary decision diagram-based procedure was used to propagate the temporal correlations from the primary inputs throughout the network. Unfortunately, the experimental data of the model were impossible to compare with experimental data of the mainstream EDA tool.

The traditional Boolean logic function has several limitations in the area optimization of logic circuits. Hence, Zhao [20] proposed an area optimization algorithm for combinational logic circuit. It divided logic circuits into several parts. Some parts utilized traditional Boolean logic for area optimization, while the other used RM logic. The proposed algorithm showed better performance than existing logic synthesis algorithms in area optimization without using any intelligence algorithm.

Bu [21] proposed a hybrid multi-valued DPSO for the minimization problem of the MPRM logic circuit. Compared with simulated annealing genetic algorithm (SAGA), this hybrid multi-valued DPSO increased the time efficiency of MPRM minimization while achieving comparable optimization results. However, the experimental data were compared only with the optimization results of SAGA and not with the optimization results of the mainstream EDA tool.

Yang [22] designed a whole annealing genetic algorithm (WAGA), which was used to search for MPRM functions for obtaining optimal circuit implementation. By combining the global searching ability of a GA and the local searching ability of SA, WAGA could achieve fast convergence. The algorithm was more effective than other GA methods in searching for the best MPRM functions. Nevertheless, experimental results were unable to show the superiority of WAGA to the EDA tool.

Many studies are available on the area optimization of MPRM logic circuits based on different intelligence algorithms. Their experimental results were mainly proposed based on intelligence algorithms. No verification method exists for comparing the area optimization results of the EDA tool and the intelligence algorithm to determine whether the intelligence algorithm is superior to the EDA tool for area optimization. Accordingly, a new verification method is designed in this study. First, a SADPSO algorithm was designed and applied in the area optimization of an MPRM logic circuit, and an area calculation equivalent model was designed. Relevant experimental data were calculated. Second, the format of the MPRM logic circuit was converted into a circuit file that can be recognized and optimized by the DC algorithm. Area optimization results of the MPRP logic circuit through the DC algorithm were calculated under the same area calculation equivalent model conditions using certain settings and constraints. Finally, the optimization results of the SADPSO and DC algorithms were compared to determine whether the intelligence algorithm outperformed the EDA tool in terms of area optimization of the logic circuit.

The remainder of this study is organized as follows. Section 3 introduces the intelligence algorithm of the SADPSO and applies it to the area optimization of an MPRM logic circuit. The area calculation model of the logic circuit is also proposed. The MCNC benchmark circuits are incorporated into the DC algorithm, and area optimization is implemented through standard cell library settings and constraints. Section 4 presents the experiment based on the MCNC benchmark circuits and analyzes the experimental data of the SADPSO algorithm. Area optimization results of the MPRM logic circuit are compared with those of the mainstream EDA tool. Section 5 concludes this study.

3. Methodology

3.1 MPRM mathematic model

The Boolean expression of n-input combinational logic circuit is:

$$f(x_{n-1},x_{n-2},...,x_0)=\sum_{i=0}^{2n-1}(a_i m_i) \tag{1}$$

where $\sum$ is the logic or operation. $a_i \in \{0,1\}$ is the coefficient of m_i [i is $(i_{n-1},i_{n-2},\cdots,i_0)$] and is a binary system. m_i can be expressed by the i^{th} term "AND" $(\dot{x}_{n-1}\dot{x}_{n-2}\cdots\dot{x}_j\cdots\dot{x}_0)$, where values of the k^{th} $\dot{x}_k$ are:

$$\dot{x}_k=\begin{cases}x_k, i_k=1\\ \bar{x}_k, i_k=0\end{cases}, 0\le k\le n-1 \tag{2}$$

One MPRM logic circuit with n-input and m-output has 3^n mixed polarities:

$$f^p(x_{n-1},x_{n-2},...,x_0)=\oplus\sum_{j=0}^{2^n-1}(b_j\pi_j) \tag{3}$$

where $(x_{n-1}, x_{n-2}, ..., x_0)$ is the input of the MPRM circuit. $p(0 \le p \le 3^n - 1)$ represents the polarity value of the MPRM and could be expressed as $(p_{n-1} p_{n-2} \cdots p_i \cdots p_0)$, where $p_i \in \{0,1,2\}$. It is a ternary system. $\oplus\sum$ is XOR; $b_j = [b_{0,j}, b_{1,j}, \cdots b_{m-1,j}]^T$ is the j^{th} jth coefficient vector of MPRM; and $\pi_j = (\dot{x}_{n-1} \dot{x}_{n-2} \cdots \dot{x}_j \cdots \dot{x}_0)$ is the jth product item of the MPRM. Values of $\dot{x}_j$ in term "AND" are shown in Table 1.

Table. 1. Value search of MPRM expansion term x_j

j_i	0	0	0	1	1	1
p_i	0	1	2	0	1	2
x_i	1	1	$\overline{x_i}$	x_i	$\overline{x_i}$	x_i

3.2 Equivalent area model

In the pipelined design of a digital integrated circuit, the combinational logic circuit is located between two triggers and generally has multiple inputs and outputs. Input signals of the combinational logic circuit arise from the output of the previous level of the trigger or the chip input end. The terminal point of the combinational logic circuit is the input of the next level of the trigger or the chip output end. The area optimization of a combinational logic circuit with multi inputs and outputs is shown in Fig. 2. The inputs of the combinational logic circuit come from the outputs of the trigger. In turn, the outputs of the combinational logic circuit are used as the inputs of the next level of the trigger. The dotted line depicts the combinational logic circuit before area optimization. SADPSO and DC are used for area optimization of the combinational logic circuit while maintaining consistent logic functions. The optimized combinational logic circuit is shown as the green area.

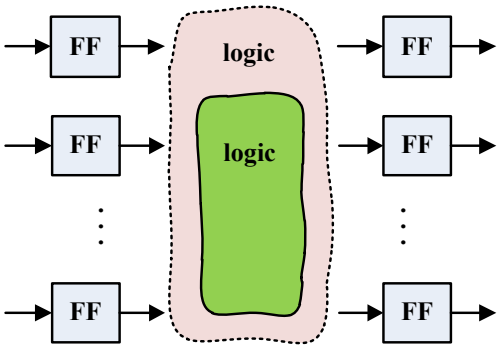

Fig. 2. Area optimization of combinational logic circuit

Different types of logic gates have dissimilar areas. The logic gate that forms the MPRM circuit is mainly composed of a phase inverter, exclusive-OR gate (XOR gate), and two-input AND gates. To construct the area calculation model, each logic gate in the MPRM circuit is hypothesized as one unit area for the statistics of the MPRM circuit after SADPSO optimization. The area of this MPRM circuit can be ascertained from the total number of logic gates and expressed as:

$$Area_1 = Count(gate) \tag{4}$$

where $Area_1$ is the number of equivalent gates after area optimization by the SADPSO algorithm, and $Count(gate)$ indicates the tallying of all logic gates. To verify the optimization effect of the SADPSO algorithm, the same combinational logic circuit was used for logic synthesis by the DC algorithm. The structure of the combinational logic circuit is restricted to the phase inverter, XOR gate, and two-input AND gates by setting certain constraints and mapping the logic gates. The number of the applied logic gates was reported through a DC software and can be expressed as:

$$v_i = (v_{i,0}, v_{i,1}, \cdots, v_{i,j}, \cdots, v_{i,n-1}) \tag{5}$$

where w is the total number of used logic gates in netlist after logic synthesis by the DC algorithm, and $Area_2$ is the total number of equivalent logic gates in the logic circuit.

3.3 SADPSO algorithm

SA, a universal search algorithm, is often used to solve the optimization problems of a combinational circuit. It can trace the search directions of the improved objective functional values through repetition, thus avoiding local minima. Suppose X_n is the initial state. SA generates one new state X_n' randomly according to the X_n field. The formula of new state transition probability based on the Metropolis criterion judgment is given by:

$$p = \begin{cases} 1, f(X_n) \ge f(X_n') \\ exp\left[\frac{f(X_n) - f(X_n')}{T_r}\right], f(X_n) < f(X_n') \end{cases} \tag{6}$$

$$T_{r+1} = \alpha \cdot T_r \tag{7}$$

where $f(X_n)$ is the cost function, and T_r is the temperature at the r iteration. Equation 7 shows the annealing function. $\alpha \in (0,1)$ denotes the temperature attenuation rate.

Particle swarm optimization (PSO) is an intelligent search algorithm based on biotic community. It is characterized by a simple solving process, few parameters, good convergence, and high robustness. PSO also has significant advantages in solving complicated combinational optimization problems. The core idea of PSO is that particles can make automatic dynamic adjustments and finally identify the optimal solution. Kennedy and Eberhart proposed the discrete PSO (DPSO) In 1997. In the DPSO algorithm, suppose that the total number of particles is m. Particle speed and position are initialized randomly. The position of the particle i in the n-dimensional space can be expressed as $x_i = (x_{i,0}, x_{i,1}, \cdots, x_{i,j}, \cdots, x_{i,n-1})$ and its flying velocity can be depicted as $v_i = (v_{i,0}, v_{i,1}, \cdots, v_{i,j}, \cdots, v_{i,n-1})$. The optimal position of the particle is calculated as $pbest_i = (pbest_{i,0}, pbest_{i,1}, \cdots, pbest_{i,j}, \cdots, pbest_{i,n-1})$, and the optimal position of all particles is $gbest_i = (gbest_0, gbest_1, \cdots, gbest_j, \cdots, gbest_{n-1})$. The updating formulas for particle velocity and position are:

$$\begin{aligned} v_{i,j}(t+1) &= w \cdot v_{i,j}(t) + c_1 \cdot r_1 (pbest_{i,j} - x_{i,j}(t)) \\ &+ c_2 \cdot r_2 \cdot (gbest_j - x_{j,j}(t)) \end{aligned} \tag{8}$$

$$x_{i,j}(t+1) = round(\frac{M}{1+exp(-v_{i,j}(t+1))} + (M\text{-}1)\cdot k\cdot random()) \quad (9)$$

where w is the inertia weight, c_1 and c_2 are the accelerated factors, r_1 and r_2 are random numbers in the range of [0,1], $round$ means the rounding operation, M is the number of value states of $x_{i,j}$, and k is a constant.

The DPSO algorithm is readily trapped in the local optimal solution. Hence, in this study, the SADPSO hybrid intelligence algorithm was proposed to optimize the searching performance. The area optimization of the combinational logic circuit based on the SADPSO algorithm proceeds as follows:

1) Parameters related to the SA and DPSO are initialized.
2) The BLIF netlist is input and converted to the corresponding format.
3) A position vector is generated randomly, and the SA initial temperature is calculated.
4) The velocity and position of the particle swarm are calculated, and $pbest_i$ and $gbest_i$ are updated.
5) The SA algorithm is started.
   (1) The optimal solution of the DPSO is used as the initial state of the SA.
   (2) A new state x'_n is generated, and whether the new state is accepted or not is determined according to Equation (6). If yes, then step (4) is performed. Otherwise, step (3) is performed again.
   (3) If the condition of convergence is met, then step (4) is performed. Otherwise, step (2) is performed again.
   (4)The optimal solution of the SA is compared with that of the DPSO, and the optimal solution of the DPSO is updated.
6) The DPSO algorithm is started, and the times of iterations, $pbest_i$ and $gbest_i$, are updated. Whether the global optimal condition is reached is determined according to the constraint. If yes, then the algorithm ends. Otherwise, recalculation and updating are repeated.
7) The algorithm ends, and the optimal value is output.

3.4 Verification method

To compare the area optimization of the MPRM combinational logic circuit with that of the SADPSO and DC algorithms, a verification method was designed. The verification process is shown in Fig. 3. MCNC benchmark circuits were used as the MPRM circuit in the verification process. First, the SADPSO algorithm was used for area optimization of the MCNC benchmark circuit, and an equivalent area was calculated according to Equation (4). Second, the format of the MCNC benchmark circuit was modified. A DC logic synthesis tool was used to impose restriction conditions such that the available logic gate types must be completely consistent with the MPRM and area constraint needs to be executed strictly. The number of equivalent logic gates was calculated according to the area optimization results of the DC logic synthesis and Equation (5). Finally, the area optimization results of the DC logic synthesis and the SADPSO were compared.

The proposed SADPSO algorithm was implemented by C++ language and compiled with VC6.0 in Windows. The area of the MCNC benchmark circuit was optimized using the SADPSO algorithm. For DC recognition and area optimization of the MCNC benchmark circuit, the circuit was transformed into the VHDL format. Each optimization was tested independently 20 times for better statistical results . The DC verification steps are as follows:

(1) The MCNC benchmark circuit was converted from the BLIF format into the VHDL format.
(2) With the use of the DC algorithm, the area of the MCNC benchmark circuit was restricted to the VHDL format to ensure that the available types of logic gates were consistent with the MPRM circuit.
(3) The area constraint was executed, and the area of the combinational logic circuit was calculated according to Equation (5).
(4) The above steps were repeated 20 times for the MCNC benchmark circuit, and the results were recorded.

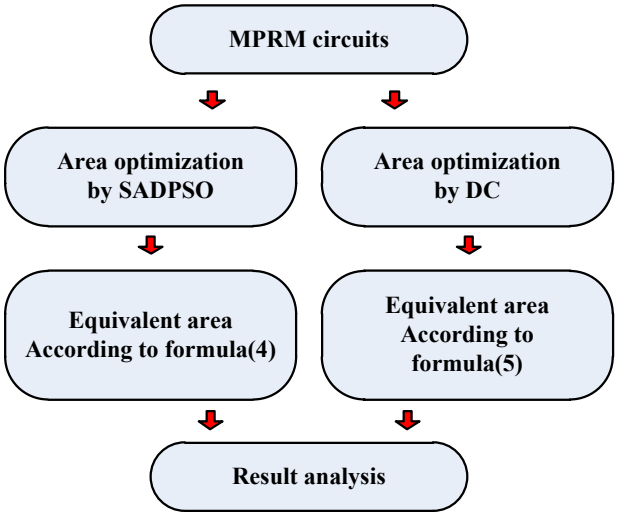

Fig. 3. Verification process

4. Result Analysis and Discussion

The SADPSO algorithm was used for the area optimization of 17 MPRM logic circuits. The same combinational logic circuit was operated independently 20 times, and the area of the optimized combinational logic circuit was calculated according to Equation (4). Then, the DC algorithm was used for area constraint and for setting to the same MCNC benchmark circuits to ensure that the logic gates covered in the netlist after logic synthesis are completely consistent with the logic gates in the MPRM circuits. Area constraint and optimization were performed using the DC algorithm. The areas of the combinational logic circuits were calculated according to Equation (5). Finally, the optimized areas of the combinational logic circuits obtained by the two algorithms were compared to verify the superority of the proposed SADPSO algorithm.

As one of the MCNC benchmark circuits, alu2 has 10 input ends and six output ends. In accordance with the features of MPRM logic circuits, available logic gates were restricted and area optimization was performed using the DC algorithm. The usage report of the logic gates is shown in Fig. 4. Dotted lines reflect the type of used logic gates and the available logic gates that were restricted into the AND gate, phase inverter, and the XOR gate respectively.

According to Equation (4), alu2 has 428 equivalent logic gates.

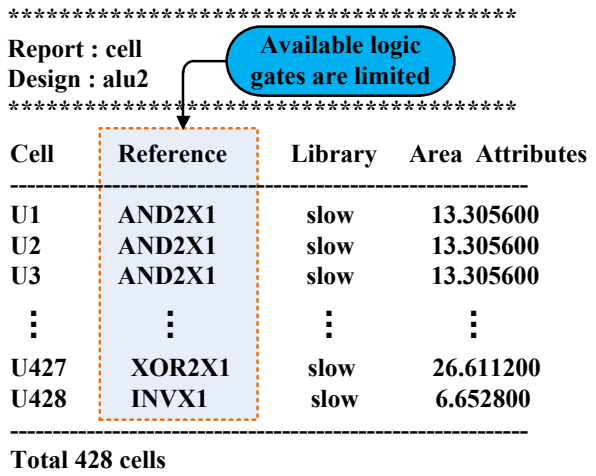

Fig. 4. Area optimization after restriction of available logic gates

The circuit after normal area optimization by DC without restriction on the available logic gates in alu2 is shown in Fig. 5. Dotted lines denotes the types of used logic gates. Any logic gate in the standard cell library can be used. According to the netlist after the logic synthesis of alu2 and Equation (5), 170 equivalent logic gates were in alu2. A comparison of the results in Figs. 4 and 5 revealed that the equivalent logic gates in alu2 increased sharply after the available logic gates were restricted. This trend occurred because, during the area optimization in DC after the restriction, the logic gates that can be used were restricted into the two-input AND gate, phase inverter, and two-input XOR gate according to the MPRM circuit. Therefore, more logic gates were needed to form the logic gate. Without a restriction on the logic gates, DC will use any logic gate to optimize the circuit according to the needs of the logic circuit, thus generating a logic circuit with a small number of equivalent logic gates.

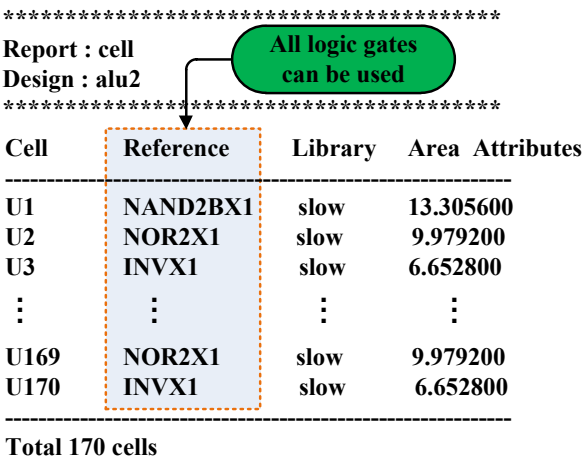

Fig. 5. Area optimization without restriction on logic gates

The area optimization process of the remaining MCNC benchmark circuits is similar to that of alu2. Area optimization results of the SADPSO and the DC algorithms to the 17 MCNC benchmark circuits are shown in Table 2, where "Circuits" is the name of MCNC benchmark circuit, "Inputs" is the number of input ports, and "Outputs" is the number of output ports in the benchmark circuits. The same benchmark circuit was optimized 20 times by the SADPSO algorithm, where "min" means the minimum quantity of equivalent logic gates calculated from Equation (4) after 20 times of optimization and "avg" is the mean quantity of equivalent logic gates. DC was applied for 20 independent area optimizations to the same benchmark circuits. The number of equivalent logic gates after logic synthesis was calculated according to Equation (5), where "min" is the minimum quantity of equivalent logic gates, "avg" is the mean quantity of equivalent logic gates, and "normal" represents the area optimization results of the logic circuit when no restrictions are placed on the available type of logic gates.

Table. 2. Area optimization of MCNC benchmark circuits

Circuits	Inputs*	Outpus*	SADPSO (quantity)		DC (quantity)		
			min	avg	min	avg	normal
alu2	10	8	401	407.6	414	421.4	170
alu4	14	8	738	749.8	776	796.6	316
b9	41	21	113	119.5	124	125.8	59
cm85a	11	3	58	63.4	50	64.8	23
comp	32	3	117	122.2	124	127.2	55
count	36	16	153	159.8	162	171.4	49
dalu	75	16	854	867.8	875	979.6	355
k2	45	45	961	977.4	1039	1048.2	479
my_adder	33	17	144	149.6	152	156.6	64
pcler8	27	17	107	107.6	111	111.2	49
pcle	19	9	61	61.8	63	63.6	24
pm1	16	13	42	45.6	52	53.4	29
t481	16	1	199	205.8	210	247.2	144
term1	34	10	209	216.8	226	234.6	139
too_large	39	3	677	690.8	700	908.8	575
ttt2	25	21	176	189.2	199	213.6	102
vda	17	40	524	535.6	556	561.8	264

* The total number of inputs and outputs.

Table 2 shows that certain differences exist between the two algorithms in terms of "min" and "avg" after area optimization of the MPRM logic circuits. This finding indicates that both SADPSO and DC algorithms exhibit unstable factors for the area optimization of the MPRM combinational logic circuit. Compared with area optimization under normal conditions, for the same MCNC benchmark circuits, equivalent logic gates in the DC algorithm after the restriction of available logic gate types increased by more than twice. Such a finding is reasonable because DC can use any logic gate to reconstruct the logic circuit under a normal pattern. The numbers of equivalent logic gates generated by the two algorithms are generally close. Specifically, the SADPSO algorithm achieves better area optimization effect for the combinational logic circuits and can save about 9.1% equivalent logic gates on average. This outcome suggests that the intelligence algorithm is better than the DC algorithm in terms of area optimization of MPRM combinational logic circuits. Our results can provide valuable references for future EDA tools design with better internal algorithm.

5. Conclusions

To verify that the intelligence algorithm is better than the DC algorithm in the area optimization of an MPRM logic circuit, a novel verification method was proposed in this study. First, a SADPSO algorithm was designed and used to optimize the area of an MPRM logic circuit. The number of equivalent logic gates after area optimization was calculated. Second, area constraints and optimization of the MPRM logic circuit were performed using the DC algorithm, and the number of equivalent logic gates was calculated. Finally, the optimization results of the MCNC benchmark circuits using the two algorithms were analyzed and compared. The following conclusions were drawn:

(1) The applied SADPSO algorithm has strong optimization ability. Among EDA tools for area optimization of the logic circuit, the DC algorithm is the most popular one used in the industry. The experiment with the MCNC benchmark circuits revealed that the SADPSO algorithm has better area optimization ability than the DC with respect to MPRM logic circuits.

(2) The proposed verification method is feasible and effective. In this study, MCNC benchmark circuits were used in the verification experiment. The area optimization results of the two algorithms were converted into the number of equivalent logic gates. The suggested verification method can determine which algorithm has better area optimization directly by comparing the hardware data of the two algorithms side by side.

In this study, the features of the MPRM logic circuits are considered comprehensively and experiment on the MCNC benchmark circuits demonstrate that the SADPSO algorithm has better area optimization ability than DC. Results also provide design guidance for EDA tools in the aspect of area optimization of logic circuits. This study investigates the experimental results of only 17 MCNC benchmark circuits, thus presenting certain limitations with regard to the types and quantities of experimental objects.

This optimization study reveales that the parameters of SADPSO algorithm and logic synthesis constraints of DC affecte the area optimization results of MPRM logic circuits to some extent. They can be analyzed by using a reasonable algorithm design and experimentation. Influencing factors should be considered in our future studies.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61404030).

References

1. Cortadella, J., Galceran-Oms, M., Kishinevsky, M., Sapatnekar, S. S., "RTL synthesis: from logic synthesis to automatic pipelining". *Proceedings of the IEEE,* 103(11), 2015, pp.2061-2075.
2. Das, A., Pradhan, S. N., "Shared reed-muller decision diagram based thermal-aware and-xor decomposition of logic circuits". *VLSI Design,* 2016(1), 2016, pp.1-14.
3. Khan, M. H., "Design of reversible synchronous sequential circuits using pseudo Reed-Muller expressions". *IEEE transactions on very large scale integration (VLSI) systems,* 22(11), 2014, pp.2278-2286.
4. Sun, W., Hou, J., "A MPRM-based approach for fault diagnosis against outliers". *Neurocomputing*, 190, 2016, pp.147-154.
5. Wang, P., Wang, Z., Rui, X., Jiang, Z., Wang, D., "Conversion algorithm for MPRM expansion". *Journal of Semiconductors*, 35(3), 2014, pp.146-151.
6. Lou, X., Yu, Y. J., Meher, P. K., "Fine-grained critical path analysis and optimization for area-time efficient realization of multiple constant multiplications". *IEEE Transactions on Circuits and Systems I: Regular Papers*, *62*(3), 2015, pp.863-872.
7. El-Maleh, A. H., Sheikh, A. T., Sait, S. M., "Binary particle swarm optimization (BPSO) based state assignment for area minimization of sequential circuits". *Applied soft computing*, 13(12), 2013, pp.4832-4840.
8. Srinivasu, B., Sridharan, K., "A Synthesis Methodology for Ternary Logic Circuits in Emerging Device Technologies". *IEEE Transactions on Circuits and Systems I: Regular Papers*, 64(8), 2017, pp.2146-2159.
9. Alexandrescu, D., Altun, M., Anghel, L., Bernasconi, A., Ciriani, V., Frontini, L., Tahoori, M., "Logic synthesis and testing techniques for switching nano-crossbar arrays". *Microprocessors and Microsystems*, 54, 2017, pp.14-25.
10. Shahsavani, S. N., Lin, T. R., Shafaei, A., Fourie, C. J., Pedram, M., "An Integrated Row-Based Cell Placement and Interconnect Synthesis Tool for Large SFQ Logic Circuits". *IEEE Transactions on Applied Superconductivity*, 27(4), 2017, pp.1-8.
11. Roberge, V., Tarbouchi, M., Labonté, G., Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Transactions on Industrial Informatics*, 9(1), 2013, pp.132-141.
12. Dai, M., Tang, D., Giret, A., Salido, M. A., Li, W. D., "Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm". *Robotics and Computer-Integrated Manufacturing*, 29(5), 2013, pp.418-429.
13. Shirazi, A., "Analysis of a hybrid genetic simulated annealing strategy applied in multi-objective optimization of orbital maneuvers". *IEEE Aerospace and Electronic Systems Magazine*, 32(1), 2017, pp.6-22.
14. Liu, Z. H., Wei, H. L., Zhong, Q. C., Liu, K., Xiao, X. S., Wu, L. H., "Parameter estimation for VSI-Fed PMSM based on a dynamic PSO with learning strategies". *IEEE Transactions on Power Electronics*, 32(4), 2017, pp.3154-3165.
15. Jiang, L., Pei, J., Liu, X., Pardalos, P. M., Yang, Y., Qian, X., "Uniform parallel batch machines scheduling considering transportation using a hybrid DPSO-GA algorithm". *The International Journal of Advanced Manufacturing Technology*, 89(5-8), 2017, pp.1887-1900.

16. Hai-Zhen, Y. U., Wang, P. J., Zhang, H. H., Kai, W., "Optimization of mprm circuits based on ternary diversity particle swarm optimization". *Acta Electronica Sinica,* 45(7), *20*17, pp.1601-1607.
17. He, Z., Xiao, L., Gu, F., Xia, T., Su, S., Huo, Z., Wang, X., "An efficient and fast polarity optimization approach for mixed polarity Reed-Muller logic circuits". *Frontiers of Computer Science*, 11(4), 2017, pp.728-742.
18. Zhang, H. H., Wang, P., GU, X., "Area optimization of fixed-polarity reedmuller circuits based on Niche genetic algorithm". *Chinese Journal of Electronics*, 20(1), 2011, pp.27-30.
19. Wang, X., Lu, Y., Zhang, Y., Zhao, Z., Xia, T., Xiao, L., "Power optimization in logic synthesis for mixed polarity Reed-Muller logic circuits". *The Computer Journal*, *58*(6), 2014, pp.1306-1313.
20. Zhao, S, S., Xia, Y. S., Zhang, J. L., Li, Q. Y., "AXIG and Area Optimization Based on Dual-Logic". *Journal of Computer-Aided Design & Computer Graphics,* 29(7), 2017, pp.1380-1388.
21. Bu, D. L., Jiang, J. H., "Hybrid multi-valued discrete particle swarm optimization algorithm for mixed-polarity Reed-Muller minimization". *Journal of Electronics and Information Technology*, 35(2), 2013, pp.361-367.
22. Yang, M., Xu, H., "Optimization of mixed polarity Reed-Muller expressions based on whole annealing genetic algorithm". In: *2011 9th IEEE International Conference on Asic,* Xiamen, China: IEEE, 2011, pp.401-404.