Postprint of Area and Delay Optimization for Ternary FPRM Circuits Based on MODCPSO Algorithm

Authors: Wang Mingbo, Wang Pengjun, Fu Qiang, Zhang Huihong

Date: 2018-05-20T00:00:00+00:00

Abstract

This paper addresses the synthesis optimization problem of area and delay for ternary fixed-polarity Reed-Muller (FPRM) logic circuits, proposing a polarity search scheme based on Multi-Objective Discrete Competitive Particle Swarm Optimization (MODCPSO). First, the MODCPSO algorithm introduces a competitive behavior mechanism that divides the population into different teams, randomly selecting two particles from each team for comparison and causing the inferior particle to update its velocity and position toward the superior particle. Simultaneously, a mutation mechanism is introduced to enable population particles to escape local optima and continue evolving. Subsequently, combining ternary FPRM polarity conversion techniques with the MODCPSO algorithm, the optimal polarity for circuit area and delay is searched. Finally, algorithm testing is implemented using MCNC Benchmark circuits in PLA format, with performance comparisons conducted against the DPSO and MODPSO algorithms. Experimental results verify the effectiveness of the MODCPSO algorithm.

Full Text

Preamble

Title: Delay and Area Optimization for Ternary FPRM Circuits Based on MODCPSO Algorithm

Authors: Wang Mingbo[1], Wang Pengjun[1]†, Fu Qiang[1,2], Zhang Huihong[1]
[1]College of Information Science and Engineering, Ningbo University, Ningbo, Zhejiang 315211, China
[2]College of Science and Technology, Ningbo University, Ningbo, Zhejiang 315212, China

Abstract: This paper addresses the synthesis optimization problem of area and delay in ternary fixed-polarity Reed-Muller (FPRM) logic circuits by proposing a polarity search scheme based on a Multi-Objective Discrete Competitive Particle Swarm Optimization (MODCPSO) algorithm. First, the MODCPSO algorithm introduces a competitive behavior mechanism that divides the population into different teams. Two particles are randomly selected from each team for comparison, and the inferior particle updates its velocity and position toward the superior particle. Simultaneously, a mutation mechanism is introduced to enable population particles to escape local optima and continue evolving. Second, the algorithm combines ternary FPRM polarity conversion techniques with MODCPSO to search for the optimal polarity that minimizes circuit area and delay. Finally, the proposed algorithm is tested using MCNC Benchmark circuits in PLA format and compared against DPSO and MODPSO algorithms. Experimental results validate the effectiveness of the MODCPSO algorithm.

Keywords: competitive behavior mechanism; multi-objective discrete particle swarm optimization (MODCPSO); ternary FPRM circuit; polarity search

0 Introduction

Boolean logic and Reed-Muller (RM) logic represent two primary forms for ternary logic functions. Compared with traditional Boolean logic circuits, RM logic-based circuits (such as arithmetic logic circuits and parity check circuits) exhibit significant advantages in area, power consumption, speed, and testability. Ternary fixed-polarity RM logic is a common logical expression form, where an n-variable RM logic function has 3 ternary FPRM expressions. Each expression varies in complexity, corresponding to different circuit area and delay characteristics. Consequently, polarity substantially impacts circuit performance metrics.

The search for optimal polarity in ternary FPRM circuits to minimize area and delay constitutes a multi-objective optimization problem. Current research primarily employs weighted coefficient methods [2,3] to transform this into a single-objective optimization problem. However, this approach suffers from several limitations: (a) weight coefficients cannot be definitively determined, and different weight settings may yield different results; (b) the method is sensitive to the shape of the Pareto frontier, making it difficult to obtain optimal solutions in concave regions; and (c) weight selection requires prior knowledge that is often unavailable. Therefore, recent studies have adopted multi-objective optimization concepts for comprehensive circuit performance optimization. For instance, reference [4] utilized Pareto dominance to optimize area and reliability of MPRM circuits, while reference [5] proposed a multi-objective discrete particle swarm algorithm for area and delay optimization in large-scale FPRM circuits.

Particle swarm optimization, genetic algorithms, and immune algorithms are effective intelligent algorithms for solving combinatorial optimization problems, widely applied in engineering [6-8]. Among these, particle swarm optimization offers robustness and fast convergence, attracting increasing attention for multi-objective problems. However, excessive traction from the global best particle often causes premature convergence. To address this, researchers have proposed various improvements. Reference [9] introduced a multi-objective quantum-behaved particle swarm algorithm using a shared learning mechanism and double-potential-well model to avoid premature convergence. Reference [10] proposed a multi-objective particle swarm algorithm based on decomposition, which divides the search space into sub-regions with different strategies to maintain diversity.

In light of these considerations, this paper proposes a multi-objective discrete particle swarm algorithm based on competitive behavior to mitigate the influence of the global best particle. This algorithm is applied to search for optimal polarity in ternary FPRM circuits for area and delay optimization. The approach maps circuit polarity to population particles, combines polarity conversion algorithms with area and delay estimation models to search for optimal polarity, and validates the algorithm using multiple MCNC Benchmark circuits.

1 Ternary Expressions and Area/Delay Estimation Models

1.1 Ternary FPRM Expressions

Any n-variable ternary logic function f(x ,x ,⋯,x) has 3 fixed polarities, with each polarity corresponding to a ternary FPRM logical expression. The ternary FPRM expression for polarity p is given by:

$$f(x_n, x_{n-1}, \dots, x_0) = \bigoplus \sum_{i=0}^{3^n-1} b_i \cdot \prod_{j=0}^{n-1} x_j^{i_j}$$

where $\bigoplus\sum$ denotes modulo-3 addition, b_i is the product term coefficient with $b_i \in \{0,1,2\}$, i is the product term index representable in ternary as $(i_n, i_{n-1}, \dots, i_0)$, p is the polarity representable as $(p_n, p_{n-1}, \dots, p_0)$, and $\prod$ denotes modulo-3 multiplication. The polarity p and product term index i determine the representation form of variable $x_j^{i_j}$, as shown in Table 1 .

When a ternary FPRM expression for a given polarity is known, another polarity expression can be obtained through polarity conversion algorithms. Reference [11] proposed a polarity conversion technique based on list algorithms that enables conversion between Boolean minterms and RM logic expressions, as well as conversion between different polarity RM logic expressions.

1.2 Area and Delay Estimation Models

In circuit design, the gate-level unit delay model is commonly used to estimate circuit delay. First, multi-input gates are decomposed into two-input modulo-3 adders and two-input modulo-3 multipliers. Circuit area is then represented by the number of two-input gates, while circuit delay is represented by the sum of transmission delays along the critical path. For a two-input gate f, the output delay t_f is:

$$t_f = \max(t_{f_a}, t_{f_b}) + t_{gate}$$

where t_{f_a} and t_{f_b} are the input delays of the two-input gate at node f.

Different decomposition methods for multi-input gates yield different circuit delays, with Huffman-like algorithms [12] being a common approach for obtaining minimal circuit delay. For the decomposed circuit network, let Mod_3A be the number of two-input modulo-3 adders, Mod_3M be the number of modulo-3 multipliers, and $num(key)$ be the total number of two-input gates on the critical path. The area and delay models for ternary FPRM circuits are therefore:

$$area = Mod\_3A + Mod\_3M$$

$$delay = \sum_{f=1}^{num(key)} t_f$$

1.3 Multi-Objective Optimization Model

Circuit polarity determines circuit area and delay. Only by obtaining the optimal polarity can circuit area and delay be comprehensively optimized. Traditional weighted coefficient methods convert this into a single-objective problem by assigning weights to area and delay. However, circuit area and delay do not necessarily follow the same trend, necessitating multi-objective optimization for comprehensive improvement.

Let $area(p)$ and $delay(p)$ denote the circuit area and delay corresponding to polarity p, respectively. Assuming p_1 and p_2 are two circuit polarities, p_1 dominates p_2 (denoted $p_1 \prec p_2$) if and only if:

$$area(p_1) \leq area(p_2) \wedge delay(p_1) \leq delay(p_2)$$

If polarity p_1 is not dominated by any other polarity, then p_1 is one of the optimal polarities satisfying the Pareto relationship between area and delay. The Pareto optimal solution set consists of all such Pareto-optimal polarities.

2 MODCPSO Algorithm

2.1 Multi-Objective Discrete Particle Swarm Algorithm

Particle swarm optimization is a swarm intelligence algorithm that simulates bird foraging behavior. During optimization, particles track individual best particles p_i and global best particles p_g to adjust their velocity and position. Assuming a D-dimensional search space, the velocity $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ and position $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ of the i-th particle are updated as:

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1 (p_{id} - x_{id}(t)) + c_2 r_2 (p_{gd} - x_{id}(t))$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

where ω is the inertia weight factor, c_1 and c_2 are learning factors, and r_1, r_2 are random numbers in $[0, 1]$.

An external archive stores the Pareto optimal solution set obtained in the current iteration, with the global best particle p_g selected from the archive via roulette wheel selection. In subsequent iterations, external archive particles are compared with population particles based on non-dominance relationships to update the archive. When the archive exceeds its capacity, particles with larger crowding distances are removed to maintain uniform distribution.

In the Multi-Objective Discrete Particle Swarm Optimization (MODPSO) algorithm, excessive traction from the global best particle often leads to premature convergence. To address this, we introduce a competitive behavior mechanism.

2.2 Multi-Objective Discrete Particle Swarm Algorithm Based on Competitive Behavior Mechanism

In MODPSO, the concentrated traction effect on each particle causes the algorithm to easily fall into local optima. Figure 1 [Figure 1: see original paper] illustrates the dynamic search process of MODPSO for minimizing a multi-peak function. Regardless of whether particles $x_1(t)$ or $x_2(t)$ are within the local optimum search space, the influence of the global best particle $p_g(t)$ and individual best particles $p_1(t)$, $p_2(t)$ ultimately traps $x_1(t)$ and $x_2(t)$ in local optima, causing premature convergence.

To overcome this, we introduce a competitive behavior mechanism. The population is divided equally into two teams, and one particle is randomly selected from each team for comparison. The inferior particle $x_l(t)$ updates its velocity and position toward the superior particle $x_w(t)$, as shown in Figure 2 [Figure 2: see original paper]. Through information exchange between teams, particle $x_l(t)$ can escape local optima, effectively avoiding premature convergence caused by concentrated effects of global and individual best particles. The velocity update formula for the inferior particle is:

$$v_{ld}(t+1) = \omega v_{ld}(t) + c_1 r_1 (x_{wd}(t) - x_{ld}(t)) + c_2 r_2 (mean(x_d) - x_{ld}(t))$$

where v_{ld} and x_{ld} are the velocity and position of the inferior particle, x_{wd} is the position of the superior particle, and $mean(x_d)$ is the average position of population particles, facilitating information acquisition from the team.

To enhance algorithm diversity, we introduce a mutation mechanism. Since particle positions use ternary encoding, gene-bit mutation is employed. A mutation probability P_m and random number $c_d \in [0, 1]$ are set. If $c_d \geq P_m$, the d-th bit of the particle is mutated: 0 becomes 1, 1 becomes 2, and 2 becomes 0.

2.3 Ternary FPRM Circuit Area and Delay Optimization Scheme Based on MODCPSO Algorithm

The algorithm proceeds as follows:

a) Initialize population particle velocities and positions, and calculate the area and delay for each particle' s corresponding circuit.

b) Compute non-dominance ranks for population particles and store Pareto non-dominated particles in the external archive (rep).

c) Divide the population into two teams and randomly select one particle from each team for comparison.

d) If a dominance relationship exists between the two particles, update the inferior particle' s velocity and position using equations (8) and (9), then calculate the corresponding circuit area and delay. If no dominance relationship exists, mutate both particles simultaneously and calculate area and delay.

e) Repeat steps c)-d) until all particles in the teams have completed comparison.

f) Compute non-dominance relationships between updated population particles and rep particles, then update the rep set.

g) If rep exceeds archive size, remove particles with larger crowding distances.

h) Repeat steps c)-g) until maximum iterations are reached, then output the area and delay of particles in rep.

The algorithm has time complexity $O(N^2)$, where N is the population size.

3 Experimental Data and Analysis

To validate MODCPSO for ternary FPRM circuit area and delay optimization, we compare it against the MODPSO algorithm from reference [5] and the Discrete Particle Swarm Optimization (DPSO) algorithm from reference [12]. Parameters are set as follows: DPSO uses weighted coefficient method with both weights set to 0.5; MODPSO and MODCPSO use multi-objective optimization with external archive size 20. Other parameters: learning factors $c_1 = c_2 = 2.0$, population size 40, maximum iterations 120, inertia weight ω ranging from 0.9 to 0.4, maximum velocity 4.0, $k = 0.2$, $M = 3$.

Since MODPSO and MODCPSO produce Pareto solution sets, we process these sets by calculating fitness for each polarity using normalization:

$$fitness_i = \frac{area_i}{\sum_{j=1}^{m} area_j} + \frac{delay_i}{\sum_{j=1}^{m} delay_j}$$

where m is the number of optimal solutions in the Pareto set. The polarity with minimum fitness is selected as the best solution for comparison.

Table 2 presents the best-polarity circuit area and delay for three algorithms running MCNC Benchmark circuits. "Name" and "input" denote circuit name and input variables; "area" and "delay" are cumulative values from five runs; "area%" and "delay%" indicate optimization rates calculated as:

$$area\% = \frac{area_1 - area_3}{area_1} \times 100\%$$

where $area_1$, $area_2$, and $area_3$ represent results from DPSO, MODPSO, and MODCPSO, respectively.

Results show MODCPSO finds polarities with smaller area and delay compared to DPSO and MODPSO. For the newtpla circuit, MODCPSO achieves area optimization rates of 18.99% and 17.67% over DPSO and MODPSO, respectively. Although area increases slightly for misex1, overall average optimization rates are 6.57% and 0.83% for area, and 4.56% and 1.12% for delay compared to DPSO and MODPSO. Variance analysis across five runs shows DPSO area/delay variances of 33.07/0.22, MODPSO 26.21/0.15, and MODCPSO 15.20/0.13, demonstrating MODCPSO' s superior robustness.

To better observe search performance, we accumulated the best area and delay solutions across five iterations for 13 test circuits. Figures 3 [Figure 3: see original paper] and 4 [Figure 4: see original paper] show the evolution curves. DPSO and MODPSO converge early and become trapped in local optima, while MODCPSO continues evolving and discovers superior polarities with smaller area and delay.

4 Conclusion

This paper proposes a polarity search scheme based on MODCPSO for ternary FPRM circuit area and delay optimization. The competitive behavior mechanism divides the population into two teams, where inferior particles update toward superior particles, avoiding premature convergence. The mutation mechanism further enables escape from local optima. Experimental results from 13

Benchmark circuits demonstrate that MODCPSO achieves better optimization efficiency and robustness compared to DPSO and MODPSO.

References

[1] Vijayakumari C K, Mythili P, James R K, et al. Optimal design of combinational logic circuits using genetic algorithm and reed-muller universal logic modules [C]// Proc of International Conference on Embedded Systems. 2014: 1-6.
[2] Wang Pengjun, Li Kangping, Zhang Hhuihong. PMGA and its application in area and power optimization for ternary FPRM circuit [J]. Journal of Semiconductors, 2016, 37 (1): 126-130.
[3] 马雪娇, 厉琼莹, 夏银水, 等. 基于双逻辑门级图形表示的功耗优化技术 [J]. 计算机辅助设计与图形学学报, 2017, 29 (3): 509-518.
[4] 卜登立, 江建慧. 基于 Pareto 支配的 MPRM 电路面积与可靠性优化 [J]. 电子学报, 2016, 11 (11): 2653-2659.
[5] 符强, 汪鹏君, 童楠, 等. 基于 MODPSO 算法的 FPRM 电路多约束极性优化方法 [J]. 电子与信息学报, 2017, 39 (3): 717-723.
[6] Wang Mingan, Shuo Feng, He Chunhui, et al. An artificial immune system algorithm with social learning and its application in industrial PID controller design [J]. Mathematical Problems in Engineering, 2017, 2017 (2017) 1-13.
[7] Ni Qingjian, Pan Qianqian, Du Huimin, et al. A novel cluster head selection algorithm based on fuzzy clustering and particle swarm optimization [J]. IEEE//ACM Trans on Computational Biology and Bioinformatics, 2017, 14 (1): 76-84.
[8] Keshanchi B, Souri A, Navimipour N J. An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification simulation, and statistical testing [J]. Journal of Systems and Software, 2017, 124 (2): 1-21.
[9] Xu Suhui, Mu Xiaodong, Dong Chai, et al. Multi-objective quantum-behaved particle swarm optimization algorithm with double-potential well and share-learning [J]. Optik: International Journal for Light and Electron Optics, 2016, 127 (12): 4921-4927.
[10] Dai Cai, Wang Yuping, Ye Miao. A new multi-objective particle swarm optimization algorithm based on decomposition [J]. Information Sciences, 2015, 325 (2015): 541-557.
[11] Yu Haizhen, Wang Pengjun, Zhang Yuejun. A ternary polarity conversion technique for fixed polarity Reed-Muller expansions [C]// Proc of International Conference on Signal Processing. 2016: 1-5.
[12] 王振海, 汪鹏君, 俞海珍, 等. 基于 PSO 算法的 FPRM 电路延时与面积优化 [J]. 电路与系统学报, 2012, 17 (5): 75-80.

Note: Figure translations are in progress. See original paper for figures.