



# A power optimization approach for mixed polarity Reed–Muller logic circuits based on multi-strategy fusion memetic algorithm<sup>\*\*#</sup>

Mengyu ZHANG<sup>1,2</sup>, Zhenxue HE<sup>†‡1,2</sup>, Yijin WANG<sup>1,2</sup>, Xiaojun ZHAO<sup>1,2</sup>,  
 Xiaodan ZHANG<sup>1,2</sup>, Limin XIAO<sup>3</sup>, Xiang WANG<sup>4</sup>

<sup>1</sup>Intelligent Agricultural Equipment Research Institute, Hebei Agricultural University, Baoding 071001, China

<sup>2</sup>Key Laboratory of Agricultural Big Data of Hebei Province, Hebei Agricultural University, Baoding 071001, China

<sup>3</sup>School of Computer Science and Engineering, Beihang University, Beijing 100191, China

<sup>4</sup>School of Electronic Information Engineering, Beihang University, Beijing 100191, China

<sup>†</sup>E-mail: hezhenxue@buaa.edu.cn

Received June 14, 2024; Revision accepted Sept. 26, 2024; Crosschecked Jan. 16, 2025

**Abstract:** The power optimization of mixed polarity Reed–Muller (MPRM) logic circuits is a classic combinatorial optimization problem. Existing optimization approaches often suffer from slow convergence and a propensity to converge to local optima, limiting their effectiveness in achieving optimal power efficiency. First, we propose a novel multi-strategy fusion memetic algorithm (MFMA). MFMA integrates global exploration via the chimp optimization algorithm with local exploration using the coati optimization algorithm based on the optimal position learning and adaptive weight factor (COA-OLA), complemented by population management through truncation selection. Second, leveraging MFMA, we propose a power optimization approach for MPRM logic circuits that searches for the best polarity configuration to minimize circuit power. Experimental results based on Microelectronics Center of North Carolina (MCNC) benchmark circuits demonstrate significant improvements over existing power optimization approaches. MFMA achieves a maximum power saving rate of 72.30% and an average optimization rate of 43.37%; it searches for solutions faster and with higher quality, validating its effectiveness and superiority in power optimization.

**Key words:** Power optimization; Multi-strategy fusion memetic algorithm (MFMA); Mixed polarity Reed–Muller (MPRM); Combinatorial optimization problem

<https://doi.org/10.1631/FITEE.2400513>

**CLC number:** TP331.1

## 1 Introduction

As the integrated circuit (IC) industry rapidly evolves alongside advancements in electronic information

technology, the significance of IC in digital systems, computer manufacturing, communication equipment, and various other domains continues to grow (Huan et al., 2021). However, with this rapid progress comes a notable issue: the escalating power of IC. This increase not only poses challenges for portable devices, causing power difficulties and chip overheating, but also impacts the cost of heat dissipation and packaging. Consequently, optimizing chip power has emerged as a pivotal factor in IC design and optimization.

Digital logic circuits can be implemented using either Boolean logic circuits, which rely on AND/OR/NOT operations, or Reed–Muller (RM) logic circuits, which use XNOR/OR or XOR/AND operations.

<sup>‡</sup> Corresponding author

\* Project supported by the National Natural Science Foundation of China (No. 62102130), Central Government Guides Local Science and Technology Development Fund Project, China (No. 226Z0201G), Natural Science Foundation of Hebei Province, China (Nos. F2020204003 and F2024204001), and Science and Technology Research Projects of Higher Education Institutions in Hebei Province, China (No. QN2024138)

# Electronic supplementary materials: The online version of this article (<https://doi.org/10.1631/FITEE.2400513>) contains supplementary materials, which are available to authorized users

ORCID: Mengyu ZHANG, <https://orcid.org/0009-0001-4414-1038>; Zhenxue HE, <https://orcid.org/0000-0001-7041-8582>

© Zhejiang University Press 2025

Numerous studies have indicated that for certain circuits (Zhao GC et al., 2024), RM logic circuits offer more significant advantages over Boolean logic circuits in terms of power, area efficiency, reliability, and testability. Consequently, RM logic circuits have garnered considerable attention as research focuses on the field of IC design and optimization (Lopera et al., 2021). The most prevalent forms of RM logic circuits are the XNOR/OR-based mixed polarity RM (MPRM) logic circuits and the XOR/AND-based fixed polarity RM (FPRM) logic circuits. For any  $n$ -variable Boolean logic circuit, there exist  $3^n$  different mixed polarities and  $2^n$  different fixed polarities. As a result, MPRM logic circuits provide a large optimization space and a great potential for improved performance compared to FPRM logic. Furthermore, varying input variables yield different circuit expressions, corresponding to distinct polarities (Liu et al., 2024). Optimizing power in MPRM circuits involves identifying the optimal polarity configuration that minimizes power, thereby posing a typical combinatorial optimization problem.

In MPRM logic circuits, the exhaustive search method effectively finds the optimal polarity for circuits with fewer input variables. However, as the number of input variables increases, the optimization space grows exponentially, making exhaustive methods time-consuming and inefficient, ultimately failing to yield optimal solutions within a reasonable timeframe. Due to their simple structure, fast search capabilities, and resistance to local optima, swarm intelligence optimization algorithms have found widespread application in MPRM circuit optimization. Zhou et al. (2022) proposed an optimizer that included global exploration and local depth exploitation and used a Huffman tree construction algorithm to seek the minimum-power-consumption FPRM circuit, although its optimization efficiency was lower when dealing with large-scale circuits. Wang X et al. (2015) proposed an improved adaptive genetic algorithm (IAGA) to optimize the best polarity traversal sequence for MPRM logic circuits, accelerating the polarity optimization, but a comparison with other classical swarm intelligence algorithms was lacking. Similarly, He et al. (2024) proposed a whale optimization algorithm (TMWOA), which employed a dual-population strategy and mutation strategy, enhancing convergence speed and enabling escape from local optima through information

exchange; however, low efficiency in large-scale optimization problems was still demonstrated. Existing MPRM logic circuit power optimization methods based on traditional swarm intelligence algorithms face challenges of slow convergence and susceptibility to local optima (Khatana et al., 2024).

In this paper, we propose a power optimization approach for MPRM logic circuits based on a multi-strategy fusion memetic algorithm (MFMA). Compared to existing power optimization approaches, our main contributions are as follows:

1. An MFMA is proposed. It comprises a global exploitation optimizer using the chimp optimization algorithm (ChOA), a local exploration optimizer based on the coati optimization algorithm based on the optimal position learning and adaptive weight factor (COA-OLA), and a population selection optimizer employing a truncation selection algorithm. The MFMA expedites convergence, enhances search accuracy, and prevents the algorithm from succumbing to local optima.

2. A power optimization approach for MPRM logic circuits is introduced, employing the MFMA to search for the optimal polarity configuration with minimal power. Notably, this paper marks the pioneering application of the ChOA and COA to RM logic circuit optimization, enhancing the search capabilities and solution quality in this domain. As far as we know, this is the first attempt to apply a memetic algorithm to optimize the power of RM circuits.

3. Experiments conducted on power using the Microelectronics Center of North Carolina (MCNC) benchmark circuit validate the efficacy and superiority of the proposed power optimization approach.

## 2 Power estimation model

Currently, the most prevalent form of ICs is complementary metal-oxide-semiconductor (CMOS) circuits. Compared to dual-gate and transmission gate logic, CMOS exhibits a stronger capability for driving loads and does not require additional logic levels, thereby simplifying the design and testing of power circuits (Li et al., 2024). Additionally, the CMOS structure offers benefits such as full-swing output voltage and symmetrical layout. However, a significant portion of the power in chips arises from dynamic power due

to the charging and discharging of load capacitances (Ju et al. 2024); the switching activity rate of the gate circuit can typically be derived from its output signal probability:

$$E_{\text{swd}}^i = 2P(g), \quad (1)$$

$$E_{\text{swd}}^i = 2P(g)(1 - P(g)), \quad (2)$$

where  $P(\cdot)$  represents the output signal probability of the logic gate, which can be obtained by combining the input signal probability with the signal probability transfer algorithm. The power of digital ICs is primarily divided into two parts: (1) dynamic power; it arises from capacitor charging and discharging during signal transitions and can be calculated using Eq. (1). Within dynamic power, switching power is the predominant factor, while short-circuit power resulting from a brief short between the input and output has a relatively minor impact. (2) Static power; it refers to the power consumed when the circuit is in a static state and not switching. This component is independent of input signals and depends solely on circuit design and operating conditions, and can be calculated using Eq. (2). Static power typically accounts for less than 1% of total power, which is significantly lower than dynamic power. Therefore, this paper focuses on calculating switching activity in dynamic logic. From Eq. (2), the MPRM circuit expression is composed of multi-input XNOR and OR operations, with power primarily arising from these components. Multi-input logic operations must be decomposed into two-input operations to calculate the power of the MPRM circuit. Consequently, its power is determined by the two-input XNOR and OR gates, as shown in Eqs. (3) and (4).

$$P(g) = 1 + 2P(x) + P(y) - P(x)P(y), \quad (3)$$

$$P(f) = P(x) + P(y) - P(x)P(y). \quad (4)$$

Meanwhile, the multi-input XNOR term and multi-input OR term need to be decomposed into a series of two-input XNOR terms and two-input OR terms, respectively. The decomposition algorithms were detailed by Qin et al. (2023), and the signal probabilities obtained by employing different decomposition algorithms would vary, as illustrated in Figs. 1 and 2, where  $\odot$  represents the XNOR operation.

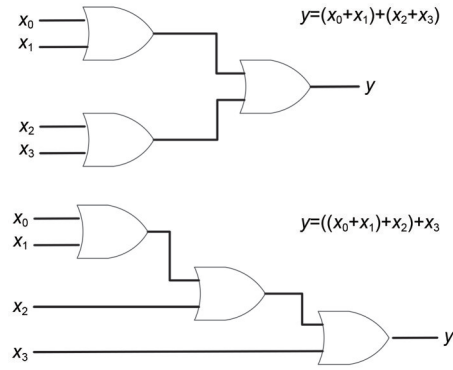


Fig. 1 OR gate decomposition approach

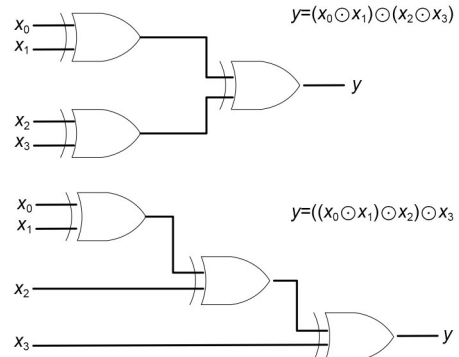


Fig. 2 XNOR gate decomposition approach

### 3 Multi-strategy fusion memetic algorithm

The memetic algorithm was first proposed by Pablo Moscato (Moscato, 1989), which is based on the concept of simulating cultural evolution for optimization purposes. Essentially, it combines a population-based global search with an individual-based local search. Since its inception, the memetic algorithm has gained popularity among researchers due to its ability to overcome the limitations of the genetic algorithm (GA) and achieve superior optimization capabilities. It has been successfully applied in various fields. The memetic algorithm adapts to various problem environments by exploring diverse solutions. It effectively reduces the search space and quickly converges to optimal or near-optimal solutions, thus enhancing the efficiency of MPRM logic circuit design. Additionally, the algorithm's flexibility allows it to better accommodate the specific constraints and requirements of MPRM logic circuits, improving power optimization efficiency. To address these challenges, we propose an MFMA.

The MFMA comprises three main components: a global exploitation optimizer employing the ChOA, a local exploration optimizer using the COA-OLA, and a population selection optimizer employing a truncation selection algorithm. The MFMA integrates these strategies to efficiently traverse the search space, balancing global exploration with local exploitation and selecting promising individuals for further exploration. Through this holistic approach, the MFMA aims to overcome the limitations of existing methods and facilitate the discovery of more effective power-optimized solutions for MPRM logic circuits. The details of the algorithm are described as follows.

Khishe and Mosavi (2020) proposed the ChOA, a novel meta-heuristic algorithm that simulates the behaviors of chimpanzee groups in attacking, driving away, obstructing, and pursuing prey. Due to its strong global search capabilities and ease of implementation, it is suitable for various optimization problems. Therefore, the ChOA is selected as the global optimization algorithm to enhance convergence speed. However, in optimizing the power consumption of MPRM logic circuits, it may fall into local optima and consume substantial computational resources. To address this, a local search algorithm can be proposed to balance exploration and exploitation, thereby improving solution quality. The COA (Dehghani et al., 2023), inspired by coati behavior, demonstrates strong adaptability and robustness, allowing for extensive exploration of the search space and improved search efficiency. Thus, using COA as the local optimization algorithm can effectively resolve issues related to local optima and resource consumption.

### 3.1 COA-OLAs

Compared to other swarm intelligence algorithms, the COA features fewer parameters, a clear division of labor among individuals, ease of implementation, and high stability. However, it is prone to becoming trapped in local optima during iterations and has a slower convergence rate (Hasanien et al., 2023). In the first stage of prey hunting, some coatis climb trees to scare the iguana until it falls, while those on the ground prepare to hunt. If the location where the iguana lands is considered the optimal position within the population, its landing spot may be chosen randomly, introducing randomness that could lead the algorithm to

a local optimum. In the second stage, when escaping from predators, a coati changes its position randomly to evade attacks. However, this strategy can result in stagnation, as it may not effectively search for more optimal solutions (Suau and Zegard, 2023). To address these shortcomings, this paper proposes a COA-OLA. This method enhances the algorithm's search accuracy and helps avoid local optima by refining the landing position of the iguana and adjusting the coati's escape strategy.

#### 3.1.1 Optimal position learning

Due to the random selection of prey positions in the initial phase of the original algorithm, the search accuracy is relatively poor. Inspired by Yıldız et al. (2023), this paper proposes an optimal position learning strategy. This strategy calculates the fitness values of the remaining coati individuals and updates the prey position by combining the best fitness individual with two randomly selected individuals. This approach enables the coati individuals to approach the prey more quickly and enhances their attack efficiency, thereby improving the algorithm's search speed. The mathematical model is as follows:

$$\begin{aligned} \text{Iguana}^G: \text{Iguana}_j^G &= x_{i,j}^1 + R_d(x_{i,j}^1 - x_{i,j}^2) \\ &\quad + (x_{i,j}^3 - \text{Iguana}_j^G), \quad (5) \\ i &= 1, 2, \dots, N_2, j = 1, 2, \dots, m, \end{aligned}$$

where  $x_{i,j}^1$  denotes the location of the optimal individual obtained through the computation of fitness values, and  $x_{i,j}^2$  and  $x_{i,j}^3$  denote two individuals randomly selected from the remaining coati population on the ground.  $R_d$  denotes the random number within the interval  $(1, d)$  dimension.  $N_2$  is the population size of coatis, and  $m$  is the dimension of the decision space.

#### 3.1.2 Adaptive weight factor

In the original algorithm's stage of escaping from predators, coati individuals randomly select a position nearby to flee. This random escape does not effectively cover the entire search space and fails to guide the coatis to explore significantly different areas, making it difficult to balance exploration and exploitation. As a result, the algorithm is prone to stagnation in local optima. Inspired by Wang WC et al. (2023), we propose an adaptive weight factor strategy.

By adjusting the inertia weight, coati individuals can choose new positions based on the distance from the predator, thereby enhancing the algorithm's local search capability. The mathematical model for this strategy is as follows:

$$w = \sin\left(\frac{\pi i}{2N_2} + \pi\right) + 1, \quad (6)$$

$$X_i^{P_2}: x_{i,j}^{P_2} = wx_{i,j}^4 + (r^2 x_{i,j}^5 - x_{i,j}^6), \quad (7)$$

$$i = 1, 2, \dots, N_2, j = 1, 2, \dots, m,$$

where  $w$  denotes the weighting factor,  $X_i^{P_2}$  represents the new position obtained by the individual coati after escaping from the predator,  $x_{i,j}^4$  denotes the location information of the optimal coati individual calculated by the fitness value, and  $x_{i,j}^5$  and  $x_{i,j}^6$  denote the location information corresponding to two randomly selected coati individuals from the coati population, respectively. The algorithm's corresponding flow is depicted in Algorithm 1.

---

#### Algorithm 1 COA-OLA

---

**Input:** Evolutionary parameters

**Output:** Optimal solution

- 1 Initialize the population
  - 2 **for**  $t=1$ : Max\_iteration **do**
  - 3     Update the position of the iguana
  - 4     Update the position of the top one-half of individuals in the population by adding optimal position learning based on the original coati optimization algorithm
  - 5     Update the position of the second half of individuals based on the original algorithm for the landing prey proximity attack strategy
  - 6     **for**  $i=1$ :  $N_2$  **do**
  - 7         Update population individual positions with adaptive weight factor strategies
  - 8     **End for**
  - 9     Retention of optimal individuals of the population
  - 10 **End for**
  - 11 **Output:** Optimal solution
- 

### 3.2 New stock selection approach

During the algorithm's iterative process, a high-quality population can achieve an even distribution of individuals and enhance diversity (Zhao BW et al., 2024). After mixing the new population obtained from local search with the original population, it is necessary to reselect individuals with better fitness to

ensure natural selection (Suau and Zegard, 2023). This selection is based on individual fitness evaluations and commonly used selection operators include tournament selection, random sampling, linear ranking selection, Monte Carlo selection, and truncation selection. Experiments conducted in this study demonstrate that using truncation selection can effectively improve population quality and accelerate convergence (Yin et al., 2024).

In truncation selection, the fitness values of the population are first calculated and sorted from best to worst, allowing only a fixed number of individuals to advance to the next iteration. The initial population  $N_1$  is generated using a global optimization algorithm, and a local search algorithm subsequently optimizes the population to create a new population  $N_2$ . After merging and sorting the two populations, the top  $N_1$  individuals with better fitness are selected for further iterations. This selection process is illustrated in Fig. 3.

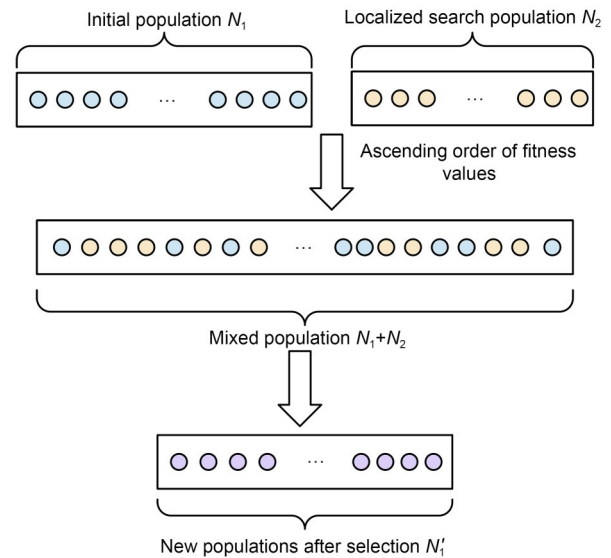


Fig. 3 Truncation of the selection process

Based on the preceding sections, the pseudo-code for MFMA is outlined in Algorithm 2.

## 4 Power optimization of MPRM logic circuits

Power optimization of MPRM logic circuits constitutes a classic combinatorial optimization challenge,

**Algorithm 2** MFMA

---

**Input:** Evolutionary parameters  
**Output:** Optimal solution

- 1 Initialize the chimp population  $N_1$
- 2 Update the position of each chimp
- 3 **for**  $l=1$ : Max\_iteration **do**
- 4     Obtain information on the top four optimal chimp individuals
- 5     Initialize the coati population  $N_2$
- 6     **for**  $t=1$ : Max\_iteration **do**
- 7         Update the position of the iguana
- 8         Update the position of the top one-half of individuals in the population by adding optimal position learning based on the original coati optimization algorithm
- 9         Update the position of the second half of individuals based on the original algorithm for the landing prey proximity attack strategy
- 10        **for**  $i=1$ :  $N_2$  **do**
- 11            Update population individual positions with adaptive weight factor strategies
- 12        **End for**
- 13        Retain the optimal individuals in the population
- 14     **End for**
- 15     Using truncation selection to obtain new populations
- 16     Update new top four optimal individuals
- 17 **End for**
- 18 **Output:** Optimal solution

---

with existing approaches often prone to issues such as susceptibility to local optima and slow convergence rates. Leveraging the power model introduced earlier, the MFMA is employed to optimize the power of MPRM logic circuits, aiming to find the polarity configuration that minimizes circuit power. The implementation steps are outlined in Algorithm 3.

## 5 Experimental results and analysis

Using Matlab R2020b for code implementation, the software operates on a Windows 10-based system, equipped with the Intel Core i7-10700 CPU, 32 GB RAM. The selected test circuit is the MCNC benchmark circuit. Performance validation of swarm intelligence optimization algorithms revolves around three key aspects: convergence speed, algorithm performance, and generalizability. Therefore, this paper conducts ablation experiments, power optimization of MPRM logic circuits, and tests on the IEEE CEC test set to validate the proposed algorithm's performance

**Algorithm 3** MPRM logic circuits power optimization

---

**Input:** Boolean circuit with  $n$  variables and evolutionary parameters  
**Output:** Optimal polarity corresponding to the MPRM logic expression that minimizes power consumption

- 1 Initialize MFMA's required parameters
- 2 Initialize the individual position and calculate the corresponding fitness value  $E_{swd}^i$
- 3 **for**  $l=1$ : Max\_iteration **do**
- 4     Obtain information on the top four optimal chimp individuals
- 5     **for**  $t=1$ : Max\_iteration **do**
- 6         Initialize the coati individual position and iguana individual position and calculate the corresponding fitness value  $E$
- 7         Execute the attacking prey phase with individual positions using an optimal position learning strategy
- 8         Execute the escape from predator phase to escape from predators using a weighting factor strategy
- 9     **End for**
- 10     Using truncation selection for population renewal
- 11     Calculate the value of individual fitness of the population  $E$
- 12     Update new top four optimal individuals
- 13 **End for**
- 14 **Output:** Optimal polarity corresponding to the MPRM logic expression that minimizes power consumption

---

across these dimensions. Given the stochastic nature of the swarm intelligence optimization algorithm, experimental results are averaged over 10 runs for each test set to ensure robustness. Additionally, it is important to consider the operating environment, parameters, and evaluation metrics of different swarm intelligence optimization algorithms. To ensure the fairness of the experiment, all parameters required for the algorithm are determined through an orthogonal experiment. The results are shown in Table 1, where the level test refers to the experimental group,  $N_2$

**Table 1** Orthogonal experiment parameters

Level test	$N_2$	$N_1$	$T$	Average result
1	26	30	35	769.22
2	26	35	45	776.68
3	26	40	40	759.58
4	28	30	45	770.47
5	28	35	40	737.27
6	28	40	35	778.21
7	30	30	40	782.99
8	30	35	35	764.89
9	30	40	45	774.34

represents the number of individuals in the coati population,  $N_1$  indicates the number of individuals in the chimpanzee population, and  $T$  denotes the maximum number of iterations for the algorithm. For the comparison algorithms—ant lion optimizer (ALO), particle swarm optimization (PSO), dandelion optimizer (DO), COA, ChOA, A-MFMA, and B-MFMA, all non-inherent parameters are unified with the proposed algorithm MFMA. Specifically, the population size is set to 35, the iteration number is set to 1000, and the number of running cycles is 10.

### 5.1 Ablation experiments

Characterized by fewer parameters and higher optimization search speed, ChOA exhibits notable features. However, when confronted with large-scale optimization problems, it is prone to occur issues such as falling into local optima and slow convergence speed. To validate the performance of each innovation, the new algorithm obtained by sequentially incorporating each innovation is compared with the original ChOA on the IEEE CEC test set. The experimental results are presented in Table 2, where “Set” denotes the name of the selected test set, “Function”

represents the name of the function randomly selected from the test set, “Standard” indicates the different evaluation metrics computed based on the experimental results, where “Ave” signifies the average value of each algorithm over 10 runs on different functions and “Best” denotes the optimal value obtained in the experiment. Additionally, A-MFMA denotes the addition of the local search algorithm COA to the original algorithm, while B-MFMA denotes the addition of the optimal position learning strategy to A-MFMA.

As depicted in Table 2, the new algorithm derived from sequentially integrating each innovation consistently outperforms ChOA across different test sets. In addition, based on the experimental averages, MFMA yields superior values, followed by A-MFMA and B-MFMA, with ChOA exhibiting the least favorable outcomes. Regarding the optimal value, defined as the minimum value obtained in the experiment, ChOA yields the largest optimal value. However, all three new algorithms incorporating different innovations demonstrate negligible differences in the optimal value, with MFMA exhibiting a general advantage. Upon analyzing the experimental results, the

**Table 2 Ablation experiment test results**

Set	Function	Standard	Value			
			ChOA	A-MFMA	B-MFMA	MFMA
2019	F3	Ave	10.1527	9.4022	7.5436	7.1943
		Best	5.3731	5.2293	3.2485	4.0357
	F6	Ave	10.8505	10.0401	9.8551	9.7078
		Best	10.1586	8.4792	8.5220	8.3931
	F8	Ave	5.3669	4.9212	4.7368	4.5605
		Best	5.0513	3.8661	4.5148	4.2269
	F9	Ave	1.5225	1.5078	1.4192	1.4077
		Best	1.2841	1.4432	1.2163	1.2949
	F10	Ave	21.4411	21.3696	21.3297	21.3272
		Best	21.3136	21.2078	21.1823	21.1388
2022	F3	Ave	660.4345	644.1511	641.4526	640.2807
		Best	640.0812	630.4283	632.6393	628.2525
	F4	Ave	930.71	891.5244	874.839	903.7881
		Best	911.9187	901.8239	884.5712	882.1141
	F5	Ave	2924.664	2143.449	2046.742	2015.967
		Best	2624.837	1792.98	1742.162	1715.962
	F9	Ave	2764.985	2571.001	2556.346	2539.089
		Best	2617.578	2492.277	2491.055	2488.957
	F10	Ave	6708.998	6282.002	5402.276	3450.28
		Best	5447.077	5374.694	2784.491	2513.652

proposed MFMA demonstrates superior performance, notably in achieving the optimal value, underscoring its superiority. The reasons for these experimental outcomes can be broadly categorized into the following three parts:

1. In scenarios with an excessively large search space, ChOA may struggle to locate the optimal value within the iterative process. To address this limitation, a new local search algorithm COA and a population selection strategy, are incorporated into the original algorithm, resulting in the development of the new algorithm A-MFMA. This integration effectively narrows down the search space, facilitating the discovery of optimal solutions while preserving the elite population. Consequently, A-MFMA demonstrates improved convergence speed compared to its predecessor.

2. Given that individual position selection in COA predominantly relies on randomness, the localized search process may lead to slow convergence. To address this issue, the new algorithm, MFMA, incorporates the optimal position learning strategy into the local search. This addition integrates optimal positions from the population into individual position selection, thereby expediting convergence speed.

3. Additionally, in the final predator escape stage of COA, an adaptive weighting strategy is introduced to adjust individual motion steps, facilitating a quicker escape from a local solution. This enhancement aims to prevent the algorithm from being trapped in local optima. Consequently, MFMA exhibits faster search capabilities and attains superior final solutions.

## 5.2 Convergence performance

To provide a more intuitive representation of convergence performance, four test functions are randomly selected and plotted based on the experimental results from Algorithm 3. Figs. 4–7 depict these plots, where the horizontal axis represents the number of iterations and the vertical axis represents the average value obtained during the iterations. It is evident that ChOA exhibits the slowest convergence, followed by A-MFMA and B-MFMA, while MFMA demonstrates the fastest convergence with superior results. This disparity can be attributed to the inherent randomness in ChOA's iterative process of position updates, resulting in a wider search space range

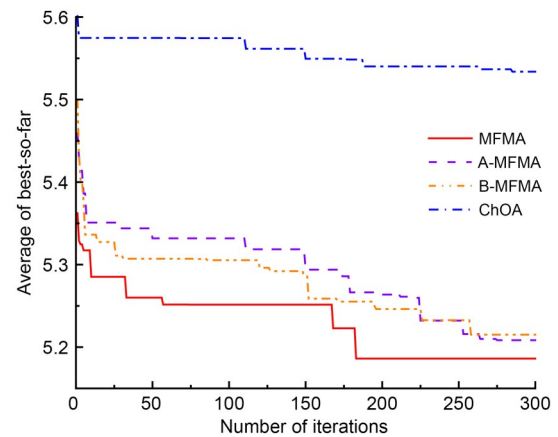


Fig. 4 Convergence performance of optimal value on CEC 2019-F8

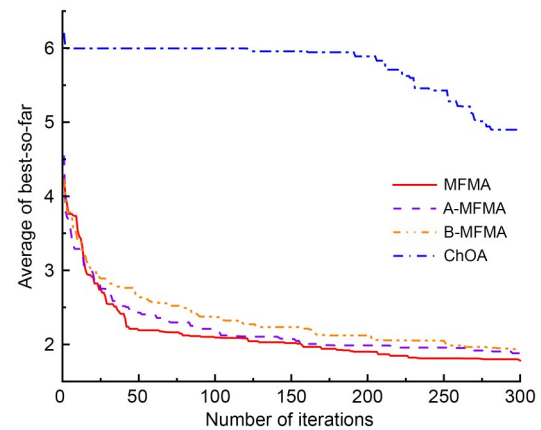


Fig. 5 Convergence performance of optimal value on CEC 2019-F9

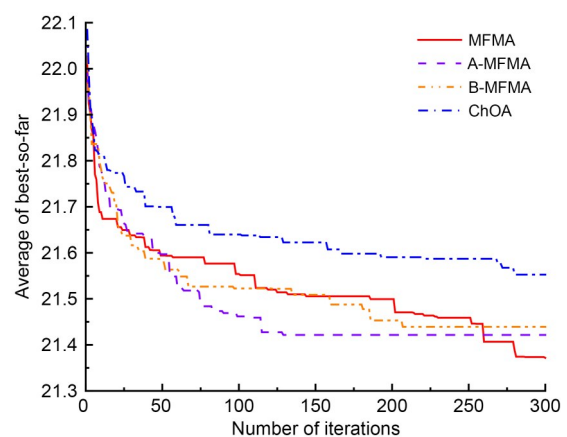
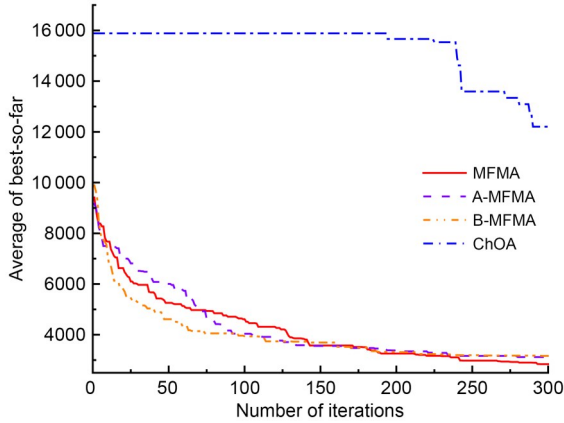


Fig. 6 Convergence performance of optimal value on CEC 2019-F10

and making it difficult to find the optimal solution. However, with the incorporation of the local search algorithm and the new population selection strategy,



**Fig. 7** Convergence performance of optimal value on CEC 2022-F5

MFMA achieves an accelerated search process, improved search accuracy, and mitigation against falling into local optima.

### 5.3 Power comparison

To validate the effectiveness of the proposed MFMA for power optimization of MPRM circuits, four algorithms—ChOA, ALO, PSO, and DO are selected for comparison with MFMA. The parameters required for MFMA have been determined through orthogonal experiments. To ensure the fairness of the experiment, the population size of all five algorithms is set to 35, and the termination condition is reaching 2000 evaluations. Twelve circuits are randomly selected from the MCNC test circuits as the experimental circuits. The experimental results are the average of 10 runs for each algorithm to mitigate the effect of randomness in swarm intelligence algorithms. As presented in Table 3, “Circuit” denotes the name of the randomly selected circuits from the MCNC test circuits. “Stand” represents the different data compared in the experiments, where “Best” indicates the optimal value obtained in the experiments (the minimum power value obtained in the experiments is considered the optimal value), “Ave” signifies the average value obtained by running the algorithm 10 times, and “Std” denotes the standard deviation of the results obtained from 10 runs of the algorithm. Additionally, “Save1” denotes the power saving rate of MFMA compared to ChOA, “Save2” represents the power saving rate of MFMA compared to PSO, “Save3” indicates the power saving rate of MFMA compared to ALO, and “Save4”

indicates the power saving rate of MFMA compared to DO. The power saving rate  $E_{\text{Save}}$  is calculated according to Eq. (8).

$$E_{\text{Save}} = \frac{E_Q - E_M}{E_Q} \times 100\%, \quad (8)$$

where  $E_Q$  and  $E_M$  denote the experimental results of the power of the three compared algorithms and MFMA, respectively.

As shown in Table 3, the circuit power results achieved by MFMA for various circuits surpass those of the other four compared algorithms. Specifically, MFMA yields an average savings of 40.75% in circuit power compared to ChOA, 59.26% compared to PSO, 47.87% compared to ALO, and 25.59% compared to DO. The optimal values indicate that PSO performs the poorest, followed by ALO and ChOA, with DO performing better, while MFMA outperforms all. In both average and standard deviation, MFMA has the best search results, proving the superiority of the proposed algorithm. The primary reasons for these experimental outcomes are as follows:

1. During the pre-iteration process of the algorithm, the initial phase involves a global search conducted by the global exploitation optimizer, which comprises ChOA. This facilitates the rapid identification of the optimal solution within the initial population. Subsequently, the location information of the optimal solution is transmitted to the local search algorithm to expedite the convergence speed of the algorithm.

2. During the local search algorithm, both the optimal position learning strategy and adaptive weight factor strategy are employed. These strategies facilitate the development of local search around the optimal individual position, thereby enhancing the algorithm’s search accuracy and preventing it from falling into local optima.

3. Following global exploitation and local exploration, the individual information of the population is selected using a truncation selection algorithm. This process retains the superior individuals from the mixed population for the next iteration, ensuring that the individuals participating in each iteration of the algorithm are the optimal ones.

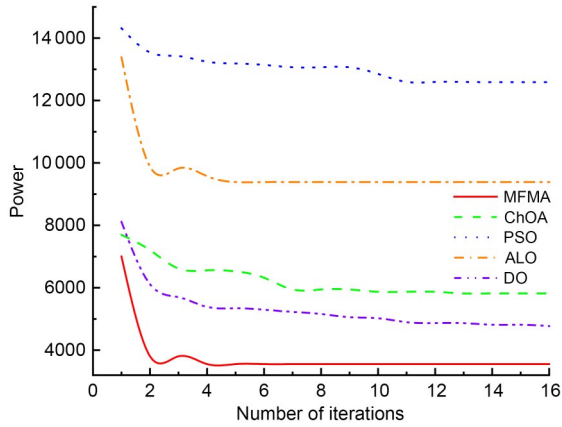
**Table 3** Experimental data on the power of four algorithms and MFMA

Circuit	Stand	Power value					Save1	Save2	Save3	Save4
		ChOA	PSO	ALO	DO	MFMA				
Sqrt8	Best	622.31	738.60	735.55	281.02	254.29				
	Ave	742.20	826.48	784.38	281.02	259.15	65.08%	68.64%	66.96%	7.78%
	Std	91.14	47.18	45.30	5.9E-14	11.27				
Prom2	Best	202.54	224.62	216.77	118.08	71.79				
	Ave	249.60	293.40	255.26	118.08	85.65	65.69%	70.81%	66.45%	27.46%
	Std	28.35	40.35	23.11	14.60	0				
Ex1010	Best	3720.57	10761.5	6514.23	4027.90	2560.75				
	Ave	5818.65	12588.4	9385.72	4704.87	3552.47	38.95%	71.78%	62.15%	24.49%
	Std	1172.51	834.04	1385.22	498.43	537.22				
Alu2	Best	138.03	138.00	135.65	136.18	89.42				
	Ave	154.17	165.86	141.60	138.06	105.52	31.56%	36.38%	25.48%	23.57%
	Std	14.75	18.72	6.74	0.68	17.89				
Clpl	Best	73.00	64.06	63.60	31.93	31.93				
	Ave	115.29	75.91	69.83	31.93	31.93	72.30%	57.94%	54.27%	0
	Std	35.85	7.39	3.33	0	0				
Br2	Best	248.84	481.70	340.79	302.18	215.26				
	Ave	322.63	968.85	555.46	487.20	304.97	5.47%	68.52%	45.10%	37.40%
	Std	48.49	260.89	138.56	161.27	51.31				
T3	Best	678.59	893.195	777.22	672.96	350.58				
	Ave	847.40	1225.41	931.30	793.21	447.42	47.20%	63.49%	51.96%	43.59%
	Std	113.82	214.48	101.43	84.89	118.65				
Newalpl1	Best	27.27	38.72	30.89	16.03	25.82				
	Ave	33.67	61.86	37.29	43.15	29.26	13.10%	52.70%	21.53%	32.19%
	Std	6.43	19.60	7.22	25.81	2.42				
Alu1	Best	137.62	134.60	134.60	128.61	72.48				
	Ave	169.06	169.32	148.43	128.61	76.01	55.04%	55.11%	48.79%	40.90%
	Std	18.98	21.29	7.83	12.27	0				
Newapla	Best	201.54	137.76	152.66	159.70	121.97				
	Ave	279.34	268.30	223.31	199.67	133.20	52.32%	50.35%	40.35%	33.29%
	Std	47.40	69.32	44.60	23.90	13.42				
Alu4	Best	3039.19	4532.91	3681.70	3275.03	1819.23				
	Ave	4011.68	5417.17	4737.93	3699.00	2560.60	36.17%	52.73%	45.96%	30.78%
	Std	562.39	831.26	473.97	550.97	355.81				
Table3	Best	5982.738	22908.89	14242.54	7962.28	8674.54				
	Ave	12341.15	31087.57	21257.02	12286.44	11589.55	6.09%	62.72%	45.48%	5.67%
	Std	2768.97	4376.48	4675.03	3069.66	2419.75				

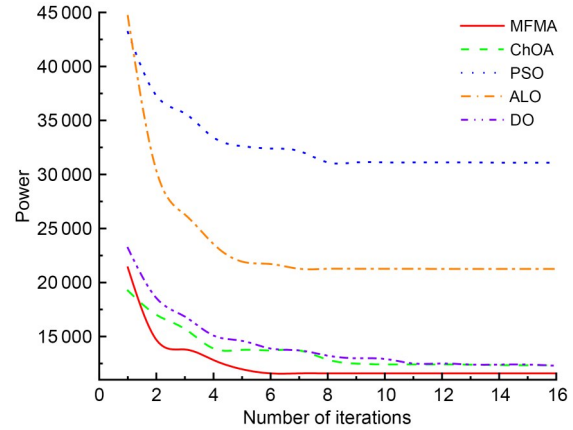
#### 5.4 Convergence comparison

To illustrate the performance of the proposed algorithms more intuitively, four circuits from Table 3 are randomly selected. The optimal solution of power obtained by each algorithm during the iteration process is calculated for these circuits. The average value of 10

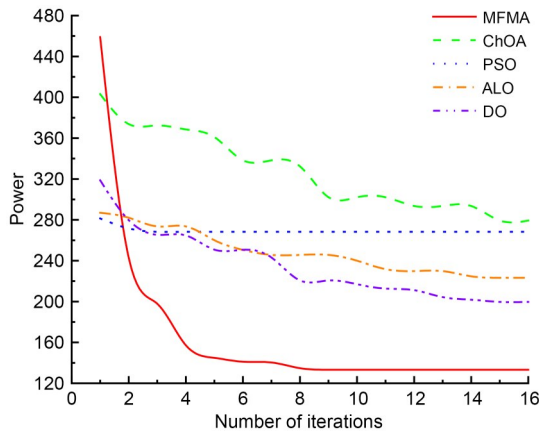
runs of individual circuits for each algorithm is then determined. Convergence curves are plotted using the acquired data, as depicted in Figs. 8–11. In these figures, the vertical axis represents the average value of power obtained from 10 runs of the algorithm. It is evident that the proposed MFMA exhibits the fastest convergence and the best optimal solution of



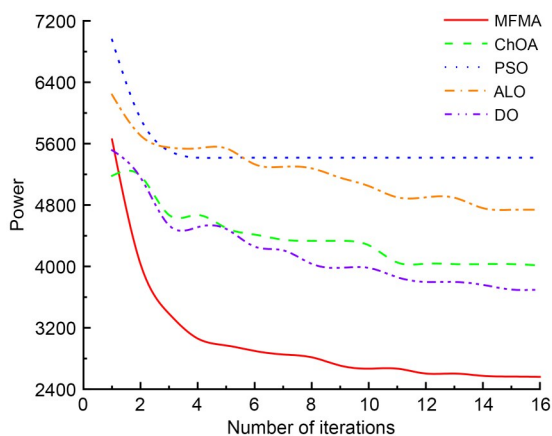
**Fig. 8** Convergence performance of optimal solution of power in Ex1010



**Fig. 11** Convergence performance of optimal solution of power in Table3



**Fig. 9** Convergence performance of optimal solution of power in Newapla



**Fig. 10** Convergence performance of optimal solution of power in Alu4

power. DO follows as the second-best solution, with ChOA and ALO follow closely behind, while PSO converges the slowest. This disparity can be attributed

to the fact that MFMA incorporates COA-OLA as a local optimization algorithm on top of the global optimization algorithm ChOA, and applies a truncation selection algorithm to preserve elite individuals during population selection.

## 6 Conclusions

The XNOR/OR-based power optimization for MPRM logic circuits belongs to a typical three-valued optimization problem. To address challenges in existing MPRM logic circuits, such as inaccurate searches and susceptibility to local optimization, we propose a power optimization approach based on the MFMA; it not only accelerates the convergence process but also enhances search accuracy, effectively resolving the three-valued combinatorial optimization problem in RM logic circuit power optimization. Experiments conducted on the MCNC benchmark circuits and the IEEE CEC function test set demonstrate that MFMA outperforms existing optimization algorithms by converging faster and identifying optimal solutions more accurately.

In the future, we plan to delve deeper into XNOR/OR-based MPRM logic circuits to develop an integrated approach for optimizing both area and power. This will involve leveraging swarm intelligence optimization algorithms to devise innovation strategies for achieving co-optimization in MPRM logic circuits. Through this research, we aim to enhance the performance of MPRM logic circuits, contributing to advancements in the field of circuit design and optimization.

## Contributors

Mengyu ZHANG designed the research and drafted the paper. Zhenxue HE designed the experiment and processed the data. Yijin WANG, Xiaojun ZHAO, and Xiaodan ZHANG helped organize the paper. Limin XIAO and Xiang WANG revised and finalized the paper.

## Conflict of interest

All the authors declare that they have no conflict of interest.

## Data availability

Data are not available due to ethical/legal/commercial restrictions.

## References

- Dehghani M, Montazeri Z, Trojovská E, et al., 2023. Coati optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems. *Knowl-Based Syst*, 259:110011. <https://doi.org/10.1016/J.KNOSYS.2022.110011>
- Hasanien HM, Alsaleh I, Alassaf A, et al., 2023. Enhanced coati optimization algorithm-based optimal power flow including renewable energy uncertainties and electric vehicles. *Energy*, 283:129069. <https://doi.org/10.1016/j.energy.2023.129069>
- He JC, He ZX, Liu J, et al., 2024. An effective power optimization approach based on whale optimization algorithm with two-populations and mutation strategies. *Chin J Electron*, 33(2):423-435. <https://doi.org/10.23919/cje.2022.00.358>
- Huan GY, Hu JB, He YF, et al., 2021. Machine learning for electronic design automation: a survey. *ACM Trans Des Autom Electron Syst*, 26(5):40. <https://doi.org/10.1145/3451179>
- Ju XX, Yuan S, Yang XS, et al., 2024. Fixed-time neurodynamic optimization algorithms and application to circuits design. *IEEE Trans Circ Syst I Regul Pap*, 71(5):2171-2181. <https://doi.org/10.1109/TCSI.2024.3349542>
- Khatana V, Saraswat G, Patel S, et al., 2024. GradConsensus: linearly convergent algorithm for reducing disagreement in multi-agent optimization. *IEEE Trans Netw Sci Eng*, 11(1): 1251-1264. <https://doi.org/10.1109/TNSE.2023.3321757>
- Khishe M, Mosavi MR, 2020. Chimp optimization algorithm. *Expert Syst Appl*, 149:113338. <https://doi.org/10.1016/j.eswa.2020.113338>
- Li HW, Chen HL, Li YX, et al., 2024. Reverse-designed photonic crystal fiber-based polarization filter with optimal performance. *Opt Laser Technol*, 168:109909. <https://doi.org/10.1016/j.optlastec.2023.109909>
- Liu AD, Fu JY, Zhan SW, et al., 2024. A policy searched-based optimization algorithm for obstacle avoidance in robot manipulators. *IEEE Trans Ind Electron*, 71(9):11262-11271. <https://doi.org/10.1109/TIE.2023.3344831>
- Lopera DS, Servadei L, Kiprit GN, et al., 2021. A survey of graph neural networks for electronic design automation. Proc ACM/IEEE 3<sup>rd</sup> Workshop on Machine Learning for CAD, p.1-6. <https://doi.org/10.1109/MLCAD52597.2021.9531070>
- Moscato P, 1989. On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms. C3P Report No. 826, California Institute of Technology, Rasadena, California, USA.
- Qin DG, He ZX, Zhao XJ, et al., 2023. Area and power optimization for fixed polarity Reed–Muller logic circuits based on multi-strategy multi-objective artificial bee colony algorithm. *Eng Appl Artif Intell*, 121:105906. <https://doi.org/10.1016/j.engappai.2023.105906>
- Suau M, Zegard T, 2023. Topology optimization with optimal design subdomain selection. *Struct Multidiscipl Optim*, 66(12):246. <https://doi.org/10.1007/s00158-023-03696-5>
- Wang WC, Tian WC, Chau KW, et al., 2023. An improved bald eagle search algorithm with Cauchy mutation and adaptive weight factor for engineering optimization. *Comput Model Eng Sci*, 136(2):1603-1642. <https://doi.org/10.32604/cmescs.2023.026231>
- Wang X, Zhang R, Wang WK, et al., 2015. Polarity searching for MPRM logic circuit based on improved adaptive genetic algorithm. Proc IEEE 12<sup>th</sup> Int Conf on Ubiquitous Intelligence and Computing and IEEE 12<sup>th</sup> Int Conf on Autonomic and Trusted Computing and IEEE 15<sup>th</sup> Int Conf on Scalable Computing and Communications and Its Associated Workshops, p.1354-1358. <https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCCom-IoP.2015.244>
- Yıldız BS, Kumar S, Panagant N, et al., 2023. A novel hybrid arithmetic optimization algorithm for solving constrained optimization problems. *Knowl-Based Syst*, 271:110554. <https://doi.org/10.1016/j.knosys.2023.110554>
- Yin H, Lyu YH, Xu LW, et al., 2025. Intelligent optimization algorithm for green IoV networks based on SSA. *IEEE Trans Veh Technol*, 74(2):2251-2260. <https://doi.org/10.1109/TVT.2023.3347744>
- Zhao BW, Chen WN, Wei FF, et al., 2024. PEGA: a privacy-preserving genetic algorithm for combinatorial optimization. *IEEE Trans Cybern*, 54(6):3638-3651. <https://doi.org/10.1109/TCYB.2023.3346863>
- Zhao GC, Zeng ZW, Wang XL, et al., 2024. Efficient ternary logic circuits optimized by ternary arithmetic algorithms. *IEEE Trans Emerg Top Comput*, 12(3):826-839. <https://doi.org/10.1109/TETC.2023.3321050>
- Zhou YH, He ZX, Chen C, et al., 2022. An efficient power optimization approach for fixed polarity Reed–Muller logic circuits based on metaheuristic optimization algorithm. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 41(12): 5380-5393. <https://doi.org/10.1109/TCAD.2022.3149720>

## List of supplementary materials

- 1 Introduction
  - 2 Preliminaries
  - 3 Multi-strategy fusion memetic algorithm
  - 4 Experimental results and analysis
- Table S1 Test set experimental results  
 Table S2 Experimental results of Wilcoxon signed-rank test  
 Fig. S1 Flowchart of ChOA  
 Fig. S2 Flow of MPRM power optimization  
 Fig. S3 Convergence performance of optimal value on different test functions  
 Fig. S4 Experimental data for box plot