Contents lists available at YXpublications

International Journal of Applied Mathematics in Control Engineering

Journal homepage: http://www.ijamce.com

Improved Red-billed Blue Magpie Algorithm and Its Application in Production Scheduling Problems

Jingru Liu[a], Lin Fu [a], Baoyan Zhao[a], Ming Ren[b], Siyi Liu[b*]

[a] *FAW Tooling Die Manufacturing CO., LTD., Changchun, Jilin 130012, China*

[b] *School of Mathematics and Statistics, Changchun University of Technology, Changchun, Jilin 130012, China*

ARTICLE INFO

Article history:
Received 24 June 2024
Accepted 30 November 2024
Available online 10 December 2024

Keywords:
Intelligent optimization
Red-billed blue magpie optimization algorithm
The production scheduling problem
Stator for electric vehicles

ABSTRACT

The Red-billed Blue Magpie Optimization (RBMO) algorithm suffers from issues such as low convergence accuracy and an imbalance between global exploration and local exploitation capabilities. To address these challenges, this paper proposes an Advanced Red-billed Blue Magpie Optimization (ARBMO) algorithm that integrates multiple strategies. Firstly, dynamic parameters are employed to dynamically adjust the search step size, enhancing the algorithm's local search capability. Secondly, to improve the algorithm's global search efficiency, Gaussian noise is randomly added to some individuals during the food-searching phase (exploration phase), aiming to redistribute the population when the algorithm gets trapped in local optima. To accelerate convergence, elite preservation and replacement mechanisms are introduced in both the food-searching and prey-attacking phases of the RBMO algorithm, ensuring that the optimal individuals are not lost due to random perturbations while also improving the overall quality of the population. Numerical experiment results verify the effectiveness of the improved algorithm. Finally, the ARBMO is applied to the production scheduling problem of the stator assembly line for electric vehicles.

1. Introduction

Population intelligence algorithms are a class of well-known meta-heuristic algorithms, which are widely used in solving complex engineering design and constrained optimization problems due to their simple principle, wide range of applicability, and the ability to quickly compute the global optimal solution [1]. Classical population intelligence algorithms such as particle swarm optimization [2] (PSO), whale optimization [3] (WOA), gray wolf optimization [4] (GWO) and simulated annealing algorithm [5] (SA) and other similar algorithms have been successfully applied to functional optimization and engineering optimal design problems. However, these algorithms still have room for further improvement in terms of optimal solution quality, problem adaptability and optimization stability. According to the No Free Lunch [6] (NFL) theorem, no single algorithm can effectively solve all optimization problems. Therefore, researchers are actively exploring new algorithms that can perform better in terms of accuracy, stability and generalization to further improve the optimization effect.

In recent years, optimization algorithms have been increasingly used in engineering due to their simple principles and other advantages. Li [7] introduced two novel and effective strategies, Lévy flight and dimension-by-dimension evaluation, into the Moth Flame Optimization Algorithm (MFO) simultaneously, which effectively improves the quality of the solution and balances the ability to search globally and develop locally. Hayyolalam [8] proposed the Black Widow Optimization Algorithm in 2020 (Black Widow Optimization Algorithm (BWOA) and applied it to solve three different types of engineering problems to verify its effectiveness. Yıldız [9], in order to improve the convergence of the original Henry gas solubility optimization algorithm (HGSO), integrated the chaotic mapping into the HGSO algorithm to improve the speed of convergence. speed, chaotic mapping was integrated into the Henry gas solubility optimization algorithm (HGSO) and compared with multiple meta-heuristic algorithms for manufacturing and diaphragm spring design problems in the automotive industry. The results show that CHGSO is a robust optimization method for obtaining optimal variables in mechanical design and manufacturing optimization problems by choosing appropriate chaotic mappings. Sadollah [10] proposed a Mine Blast Algorithm (MBA) applied to constrained optimization and engineering design problems. In the application MBA requires less number of function evaluations and

* Corresponding author.
E-mail addresses: liusiyi@stu.ccut.edu.cn (SY. Liu)
Digital Object Identifiers: https://doi.org/10.62953/IJAMCE.475026

provides better results in most cases.

Red-billed Blue Magpie Optimize [11] (RBMO) is a newly proposed meta-heuristic algorithm. RBMO is proposed to be used in UAV path planning problems and engineering setup problems, and is compared with three widely accepted algorithms to show its advantages in solving practical problems [12-14]. However, RBMO suffers from low convergence accuracy and imbalance between global exploration capability and local exploitation capability. In order to improve the optimization search accuracy of the RBMO algorithm and at the same time extend its application scope in engineering optimization and design problems, this paper proposes an algorithm of red-billed blue magpie incorporating multi-strategy (ARBMO).

In ARBMO, firstly, the search step size is dynamically adjusted using dynamic parameters to improve the local search capability of the algorithm. Second, in order to improve the global search efficiency of the algorithm, Gaussian noise is randomly added to some individuals during the food search phase (exploration phase) with a view to redistributing the population when the algorithm falls into local optimality. Finally, in order to accelerate the convergence of the algorithm, an elite preservation and replacement mechanism is introduced to the food-seeking and prey-attacking phases of the RBMO algorithm to ensure that the optimal individuals will not be lost due to random perturbations, and at the same time improve the overall quality of the population.

The main structure of this paper is as follows, Chapter 2, mainly introduces the basic theory of the optimization algorithm of the red-billed blue magpie, Chapter 3, gives the specific improvement method, Chapter 4, tests the performance of the algorithm, Chapter 5, tests the performance of the algorithm for specific engineering problems, and, finally, gives the conclusions and summarizes in Chapter 6.

2. Red-billed Blue Magpie Optimization Algorithm

Red-billed Blue Magpie Optimization Algorithm (RBMO), is inspired by the cooperative and efficient foraging behavior of red-billed blue magpies. When foraging, red-billed blue magpies use jumping, walking on the ground and searching on tree branches to find food. Early in the morning and late in the evening, red-billed blue magpies are active and often gather in small flocks of 2 to 5 or even more than 10 and hunt cooperatively. For example, if a magpie finds a fruit or an insect, it will attract other members to share it. This allows them to work together to capture large insects or small vertebrates, and group action can help them overcome the defense mechanisms of their prey. In addition, red-billed blue magpies store some food for later use. To prevent theft by other birds or animals, they hide their food in places such as tree holes, tree forks, rock crevices, etc. RBMO has developed a mathematical model of the red-billed blue magpie by simulating its searching, chasing, prey attacking, and food storage behaviors.

2.1 Initialization

As with most algorithms, the RBMO candidate solutions are generated by Eq. (1). They are randomly generated within the constraints of the given problem and need to be updated after each iteration.

$$x_{i,j} = (ub - lb) \times Rand_1 + lb \tag{1}$$

where ub and lb are the upper and lower bounds of the problem, respectively, and $Rand_1$ denote random numbers from 0 to 1.

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & x_{1,\dim-1} & x_{1,\dim} \\ x_{2,1} & \cdots & x_{2,j} & \cdots & x_{2,\dim} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n-1,1} & \cdots & x_{n-1,j} & \cdots & x_{n-1,\dim} \\ x_{n,1} & \cdots & x_{n,j} & x_{n,\dim-1} & x_{n,\dim} \end{bmatrix} \tag{2}$$

where X denotes the location of the search agent generated by Eq. (2), n denotes the population size, and dim denotes the problem-solving dimension.

2.2 Mathematical model of RBMO

2.2.1 Searching for food

To improve foraging efficiency, red-billed blue magpies usually move in small groups (2 to 5 individuals) or clusters (10 or more individuals). They will use a variety of techniques such as hopping on the ground, walking or searching for food resources in trees. This adaptability and flexibility allow red-billed blue magpies to employ different hunting strategies depending on environmental conditions and available resources, ensuring an adequate food supply. Equation (3) is used when exploring food in small groups, and Equation (4) is used when searching in clusters.

$$\mathbf{X}^i(t+1) = \mathbf{X}^i(t) + \left(\frac{1}{p} \times \sum_{m=1}^{p} \mathbf{X}^m(t) - \mathbf{X}^{rs}(t) \right) \times \mathrm{Rand}_2 \tag{3}$$

where t denotes the current iteration number, $\mathbf{X}^i(t+1)$ denotes the location of the i th new search agent, denotes the number of red-billed blue magpies hunted by a randomly selected group (between 2 and 5) of all search individuals, $\mathbf{X}^m$ denotes the m th randomly selected individual, $\mathbf{X}^i$ denotes the i th individual, and $\mathbf{X}^{rs}$ denotes the randomly selected search agent in the current iteration.

$$\mathbf{X}^i(t+1) = \mathbf{X}^i(t) + \left(\frac{1}{q} \times \sum_{m=1}^{q} \mathbf{X}^m(t) - \mathbf{X}^{rs}(t) \right) \times \mathrm{Rand}_3 \tag{4}$$

where q denotes the number of search agents for the cluster to explore for food, between 10 and n. Again, randomly selected from the entire population.

2.2.2 Attacking prey

Red-billed blue magpies show a high degree of hunting skill and cooperation when pursuing prey. They use strategies such as rapid pecking, hopping for food or flying to catch insects. In the case of small group actions, the main target is usually small prey or plants. The corresponding mathematical model is shown in Equation (5). When red-billed blue magpies act in flocks, they are able to collectively target larger prey, such as large insects or small vertebrates. The mathematical representation of this behavior is shown in Equation (6). This predatory hunting behavior highlights the diverse strategies and skills possessed by the red-billed blue magpie, making it a versatile predator adept at successfully obtaining food in a variety of situations.

$$\mathbf{X}^i(t+1) = \mathbf{X}^{food}(t) + CF \times \left(\frac{1}{p} \times \sum_{m=1}^{p} \mathbf{X}^m(t) - \mathbf{X}^i(t) \right) \times Randn_1 \tag{5}$$

$$\mathbf{X}^i(t+1) = \mathbf{X}^{food}(t) + CF \times \left(\frac{1}{q} \times \sum_{m=1}^{q} \mathbf{X}^m(t) - \mathbf{X}^i(t) \right) \times Randn_2 \tag{6}$$

where $\mathbf{X}^{food}(t)$ denotes the location of the food, $CF = \left(1 - \left(\frac{t}{T}\right)\right)\left(2 \times \frac{t}{T}\right)$, $Randn$ denotes the random number used to generate the standard normal distribution (mean 0, standard deviation 1).

2.2.3 Food Storage

In addition to searching for and attacking food, red-billed blue magpies store excess food in tree holes or other hidden places for future consumption, ensuring a steady supply in the event of a food shortage. This process retains solution information and helps the individual to find the global optimum. The mathematical model for this process is shown in Equation (7).

$$\mathbf{X}^i(t+1) = \begin{cases} \mathbf{X}^i(t) & \text{if } \text{fitness}_{old}^i > \text{fitness}_{new}^i \\ \mathbf{X}^i(t+1) & \text{else} \end{cases} \tag{7}$$

where fitness_{old}^i and fitness_{new}^i denote the fitness values before and after updating the position of the first red-billed blue magpie, respectively.

In summary, in RBMO, the optimization process begins with the generation of a set of random candidate solutions (called populations). The search strategy of RBMO explores suitable locations by repeating trajectories, either for suboptimal or optimal solutions. The food storage phase enhances the exploration and exploitation capabilities of RBMO. Finally, the RBMO search process ends when the end criteria are met. It is worth noting that the population size of red-billed blue magpies varied during the hunt, but the total number of small populations and clusters remained similar. In this paper, we propose a coefficient for balancing the populations by setting it to 0.5. The pseudo-code for RBMO is provided in Algorithm 1.

Algorithm 1

```
Start
Initialize parameters (T, n, ε etc.)
Start of iteration
  Calculate the fitness value for each search question
  Update the optimal solution
  Exploration phase:
    If rand < ε
      Update the red-billed blue magpie position using equation (3)
    Else
      Update the red-billed blue magpie position using equation (4)
  Development phase:
    If rand < ε
      Use formula (5) to update the red-billed blue magpie position
    Else
      Update red-billed blue magpie location using formula (6)
  Update X^food(t) and complete food storage using equation (7)
End iteration
Enter the optimal solution
End
```

3. Improvement of the algorithm part

Red-billed blue magpie optimizer (RBMO) is a novel meta-heuristic algorithm (intelligent optimization algorithm) inspired by the cooperative and efficient predatory behavior of red-billed blue magpies. In recent years, population intelligent optimization algorithms are widely used to solve complex optimization problems. In this context, the optimization algorithm based on the foraging behavior of red-billed blue magpies (RBMO, Red-Billed Magpie Optimization) has shown good performance. However, there is still room for improvement in the RBMO algorithm in terms of the balance between global exploration capability and local exploitation capability. The improved version of the algorithm, ARBMO (Adaptive Red-Billed Magpie Optimization), is based on RBMO and optimized for key mechanisms to further enhance the performance of the algorithm. The following is a detailed description of the improvements and advantages of ARBMO over RBMO from various perspectives. The improvement of this algorithm starts with the introduction of dynamic adjustment, which can enhance the search and convergence ability at different stages of the algorithm.

3.1 Improvement Strategies

3.1.1 Dynamic parameter adjustment

First of all, the search step size in the red-billed blue magpie is dynamically adjusted so that it gradually decreases with the number of iterations to improve the algorithm's local search ability.

In the RBMO algorithm, key parameters such as CF

(contraction factor) are calculated using a fixed formula:

$$CF=\left(1-\left(\frac{t}{T}\right)\right)^{\left(2\times\frac{t}{T}\right)} \tag{8}$$

Although the formulation varies with the current number of iterations, its form is relatively fixed and does not take into account the specific needs of exploration and exploitation at different optimization stages. Furthermore, the Epsilon parameters of the RBMO remain constant and cannot be dynamically adapted to different stages of the algorithm. Therefore, the algorithm for RBMO can be dynamically adapted with the increase of the number of iterations, and the $Epsilon$ and CF in the algorithm are improved as follows:

$$Epsilon=0.5*\left(1-\mathrm{t/T}\right) \tag{9}$$

As the number of iterations increases, $Epsilon$ decreases, thus enhancing the randomness and exploration of the population in the early stages of the algorithm and improving the accuracy of the local search in the later stages.

$$CF=\left(1-t/T\right)^2 \tag{10}$$

The tuning makes CF shrink more significantly in the later stages of the algorithm, which helps to improve convergence accuracy. Dynamic parameter tuning enhances the adaptability of the algorithm in different optimization stages, making the global search and local development more coordinated.

3.1.2 Mutation operations to enhance population diversity

The single update mechanism of RBMO makes RBMO rely only on the population update formula for population search without introducing additional mechanisms to enhance the diversity of solutions. This may cause the population to fall into local optimality in the complex search space and reduce the global search efficiency. For the original RBMO algorithm searching for food phase (exploration phase) randomly adding Gaussian noise to some individuals.

$$\begin{aligned}\mathbf{X}^i(t+1)=\mathbf{X}^i(t)+0.1\times\left(X_{\max}-X_{\min}\right)\\ \times randn(1,D)\end{aligned} \tag{11}$$

RBMO is based on the predatory behavior of the red-billed blue magpie bird in nature, where Gaussian noise is introduced in the food search phase and the mutation operation is triggered with a probability of 20%, which is able to redistribute the population when the search falls into a local optimum, increasing the ability to jump out of the local optimum.

3.1.3 Elite preservation and replacement mechanism

In the algorithm, the elite preservation and replacement mechanism is an important strategy to enhance the search efficiency and convergence performance, which is used to ensure that the optimal individuals will not be lost due to random perturbations, and at the same time, improve the overall quality of the population. The ARBMO algorithm incorporates the elite preservation and replacement mechanism, that is, at the end of each round of iteration, the best 10% individuals in the population are selected as the elites, and then these elites replace the worst-performing individuals in the population. This means that at the end of each iteration, the 10% best performing individuals in the population are selected as elites, and these elites replace the worst performing individuals in the population. The elite preservation mechanism ensures the inheritance of the optimal solutions in the population and improves the overall quality of the population by replacing the poor solutions, which helps accelerate the convergence.

ARBMO improves the updating formula in both exploratory and development phases to make the search behavior more flexible.

In the exploratory phase: two different sizes of individual population means are introduced: the large-scale individual population mean $Xpmean$ n: represents local properties. The large-scale population means $Xqmean$: represents the global characteristics.

In the development phase, ARBMO uses formulas with random perturbations to further improve the local search accuracy:

$$\begin{aligned}\mathbf{X}^i(t+1)=X_{food}+CF\times(Xpmean\\ -\mathbf{X}^i(t+1))\times randn(1,D)\end{aligned} \tag{12}$$

ARBMO further enhances the optimization capability through various improvements in dynamic parameter tuning, elite saving and replacement, and formula flexibility based on RBMO. Especially in complex multi-peak problems, ARBMO shows stronger global search capability and potential to jump out of local optimums. These improvements give it a broader application prospect in solving complex optimization problems.

4. Experimental Analysis

4.1 Test Functions

Benchmark test function is one of the important ways to evaluate the feasibility, effectiveness and stability of the optimization algorithm. In this paper, the CEC2017 test function set is selected to verify the convergence accuracy, convergence speed and stability of the algorithm through the test functions. The test functions are visualized to clearly understand the nature and characteristics of each function. The test functions are shown in the following table:

Table 1 CEC2017 test functions

Category	No.	Functions	Minimum
	1	Shifted and Rotated Bent Cigar Function	100
Unimodal Functions	2	Shifted and rotated sum of different power function	200
	3	Shifted and rotated Zakharov function	300
Simple multimodel functions	4	Shifted and rotated Rosenbrock’s function	400

	5	Shifted and rotated Rastrigin's function	500
	6	Shifted and rotated expanded Scaffer's F6 function	600
	7	Shifted and rotated Lunacek Bi_Rastrigin function	700
	8	Shifted and rotated non-continuous Rastrigin's function	800
	9	Shifted and rotated Levy function	900
	10	Shifted and rotated Schwefel's function	1000
Hybrid functions	11	Hybrid function 1 (N=3)	1100
	12	Hybrid function 2 (N=3)	1200
	13	Hybrid function 3 (N=3)	1300
	14	Hybrid function 4 (N=4)	1400
	15	Hybrid function 5 (N=4)	1500
	16	Hybrid function 6 (N=4)	1600
	17	Hybrid function 6 (N=5)	1700
	18	Hybrid function 6 (N=5)	1800
	19	Hybrid function 6 (N=5)	1900
	20	Hybrid function 6 (N=6)	2000
Composition functions	21	Composition Function 1 (N=3)	2100
	22	Composition Function 2 (N=3)	2200
	23	Composition Function 3 (N=4)	2300
	24	Composition Function 4 (N=4)	2400
	25	Composition Function 5 (N=5)	2500
	26	Composition Function 6 (N=5)	2600
	27	Composition Function 7 (N=6)	2700
	28	Composition Function 8 (N=6)	2800
	29	Composition Function 9 (N=11)	2900
	30	Composition Function 10 (N=12)	3000

4.2 Algorithm Parameter Setting

In order to further verify the performance of ARBMO, this paper compares it with eight other optimization algorithms, including: Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), Particle Swarm Optimization (PSO), Sparrow Search Algorithm (SSA), and the original Red-billed blue magpie optimizer (RBMO). In order to ensure the fairness of each algorithm, we set the population number of each algorithm to 30 and the maximum number of iterations to 500. In order to avoid the randomness of the results of a single experiment, each algorithm is run 50 times to obtain the average value (Ave) and standard deviation (Std) of the objective function, and Ave is used to reflect the convergence accuracy of the algorithm and Std is used to reflect the stability and robustness of the algorithm. The detailed parameter settings of various algorithms are shown in Table 2 below:

Table 2 Algorithm parameters information

Algorithm	Year	Parameter
PSO	1995	$c1 = c2 = 2$
GWO	2014	$\alpha = 0.2$
WOA	2016	α linearly decreasing from 2 to 0 $\alpha 2$ linearly decreasing from -1 to -2
SSA	2020	$ST = 0.8, PD = 0.2, SD = 0.1$
RBMO	2024	$p = [2,5], q = [10, n], CF = [0,1], \varepsilon = 0.5$

4.3 Analysis of Experimental Results

The running results of ARBMO with other five algorithms on the test function are shown in Table 3 below and the convergence images are shown in Figures 1-4 below.

Table 3 Comparison results of test functions

F		WOA	SSA	PSO	GWO	RBMO	ARBMO
F1	Std	1.23E+08	3.34E+03	1.93E+03	1.05E+08	2.40E+00	4.25E+03

	Ave	9.38E+07	4.08E+03	1.95E+03	3.37E+07	1.01E+02	4.57E+03
F2	Std	3.81E+07	1.13E+04	0.00E+00	1.68E+07	1.80E-01	4.74E+01
	Ave	2.70E+07	1.14E+04	2.00E+02	1.02E+07	2.00E+02	2.30E+02
F3	Std	6.39E+03	1.50E-01	8.00E-02	3.74E+03	0.00E+00	0.00E+00
	Ave	7.65E+03	3.00E+02	3.00E+02	4.10E+03	3.00E+02	3.00E+02
F4	Std	4.75E+01	1.43E+01	1.18E+01	2.32E+01	5.90E-01	1.56E+01
	Ave	4.56E+02	4.09E+02	4.06E+02	4.25E+02	4.00E+02	4.09E+02
F5	Std	1.82E+01	1.39E+01	1.30E+01	1.08E+01	4.19E+00	7.55E+00
	Ave	5.55E+02	5.25E+02	5.44E+02	5.22E+02	5.11E+02	5.11E+02
F6	Std	1.61E+01	1.37E+01	1.30E+01	2.05E+00	5.40E-01	7.70E-01
	Ave	6.40E+02	6.20E+02	6.13E+02	6.01E+02	6.00E+02	6.00E+02
F7	Std	2.78E+01	1.61E+01	8.01E+00	1.13E+01	5.20E+00	8.24E+00
	Ave	7.84E+02	7.38E+02	7.27E+02	7.34E+02	7.20E+02	7.27E+02
F8	Std	1.95E+01	1.14E+01	6.70E+00	9.08E+00	4.36E+00	5.87E+00
	Ave	8.44E+02	8.28E+02	8.20E+02	8.17E+02	8.09E+02	8.16E+02
F9	Std	3.88E+02	8.51E+01	1.23E+02	2.24E+01	8.00E-02	1.86E+00
	Ave	1.51E+03	9.75E+02	9.49E+02	9.20E+02	9.00E+02	9.00E+02
F10	Std	3.70E+02	2.49E+02	3.08E+02	3.28E+02	2.38E+02	2.51E+02
	Ave	2.27E+03	1.89E+03	1.99E+03	1.62E+03	1.59E+03	1.57E+03
F11	Std	1.04E+02	9.13E+01	2.67E+01	2.67E+01	2.91E+00	5.66E+01
	Ave	1.25E+03	1.21E+03	1.14E+03	1.14E+03	1.10E+03	1.13E+03
F12	Std	5.38E+06	4.79E+06	1.51E+04	1.02E+06	1.02E+02	1.49E+04
	Ave	4.81E+06	3.87E+06	2.03E+04	9.01E+05	1.39E+03	1.83E+04
F13	Std	1.43E+04	1.73E+04	6.42E+03	7.74E+03	1.03E+01	5.53E+03
	Ave	1.87E+04	1.96E+04	1.06E+04	1.18E+04	1.31E+03	2.65E+03
F14	Std	1.55E+03	2.68E+03	2.19E+03	2.06E+03	8.09E+00	1.46E+01
	Ave	2.79E+03	3.08E+03	2.88E+03	3.33E+03	1.42E+03	1.43E+03
F15	Std	6.91E+03	7.18E+03	3.44E+03	6.77E+03	2.95E+00	3.26E+01
	Ave	9.46E+03	8.26E+03	4.39E+03	6.59E+03	1.50E+03	1.50E+03
F16	Std	1.96E+02	1.44E+02	1.36E+02	1.71E+02	3.67E+01	7.61E+01
	Ave	1.94E+03	1.81E+03	1.87E+03	1.79E+03	1.62E+03	1.67E+03
F17	Std	4.68E+01	4.18E+01	3.55E+01	3.01E+01	2.84E+01	3.07E+01
	Ave	1.81E+03	1.79E+03	1.77E+03	1.77E+03	1.75E+03	1.75E+03
F18	Std	1.13E+04	1.29E+04	1.13E+04	1.62E+04	9.56E+00	3.46E+01
	Ave	1.50E+04	2.15E+04	1.46E+04	2.92E+04	1.81E+03	1.81E+03
F19	Std	3.03E+05	6.68E+03	4.77E+03	6.87E+04	1.33E+00	2.03E+01
	Ave	1.78E+05	9.53E+03	5.47E+03	2.74E+04	1.90E+03	1.91E+03
F20	Std	9.45E+01	8.36E+01	6.16E+01	7.18E+01	4.50E+01	4.50E+01
	Ave	2.23E+03	2.15E+03	2.12E+03	2.11E+03	2.05E+03	2.06E+03
F21	Std	3.97E+01	5.03E+01	6.37E+01	2.22E+01	2.92E+01	4.47E+01
	Ave	2.35E+03	2.30E+03	2.31E+03	2.32E+03	2.31E+03	2.30E+03
F22	Std	4.09E+02	1.25E+02	1.93E+02	1.32E+01	1.77E+00	1.27E+01
	Ave	2.42E+03	2.32E+03	2.34E+03	2.31E+03	2.30E+03	2.30E+03
F23	Std	2.22E+01	1.07E+01	3.66E+01	1.02E+01	5.52E+00	7.79E+00
	Ave	2.67E+03	2.63E+03	2.71E+03	2.62E+03	2.61E+03	2.62E+03
F24	Std	6.70E+01	4.68E+01	1.44E+02	1.18E+01	6.17E+01	6.38E+01

	Ave	2.77E+03	2.74E+03	2.76E+03	2.75E+03	2.73E+03	2.73E+03
F25	Std	4.75E+01	6.62E+01	2.26E+01	1.47E+01	2.23E+01	2.41E+01
	Ave	2.98E+03	2.91E+03	2.93E+03	2.94E+03	2.93E+03	2.93E+03
F26	Std	5.50E+02	7.60E+01	3.94E+02	2.95E+02	1.84E+02	2.55E+02
	Ave	3.53E+03	2.90E+03	3.07E+03	3.05E+03	2.96E+03	3.01E+03
F27	Std	4.48E+01	3.84E+00	6.58E+01	1.92E+01	1.25E+01	2.11E+01
	Ave	3.16E+03	3.09E+03	3.16E+03	3.10E+03	3.10E+03	3.10E+03
F28	Std	1.81E+02	1.46E+02	7.94E+01	1.02E+02	1.82E+02	1.50E+02
	Ave	3.44E+03	3.30E+03	3.22E+03	3.39E+03	3.38E+03	3.35E+03
F29	Std	1.10E+02	6.24E+01	7.65E+01	4.91E+01	3.08E+01	5.46E+01
	Ave	3.40E+03	3.25E+03	3.25E+03	3.21E+03	3.16E+03	3.20E+03
F30	Std	2.01E+06	1.16E+06	4.28E+04	1.08E+06	4.76E+05	4.39E+05
	Ave	1.87E+06	7.50E+05	4.78E+04	1.03E+06	4.55E+05	2.80E+05

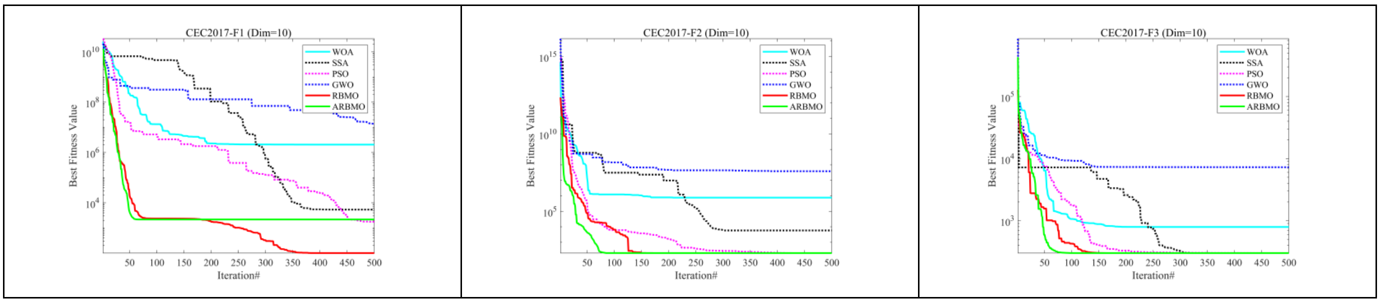


Fig. 1 Convergence curve of Unimodal Functions

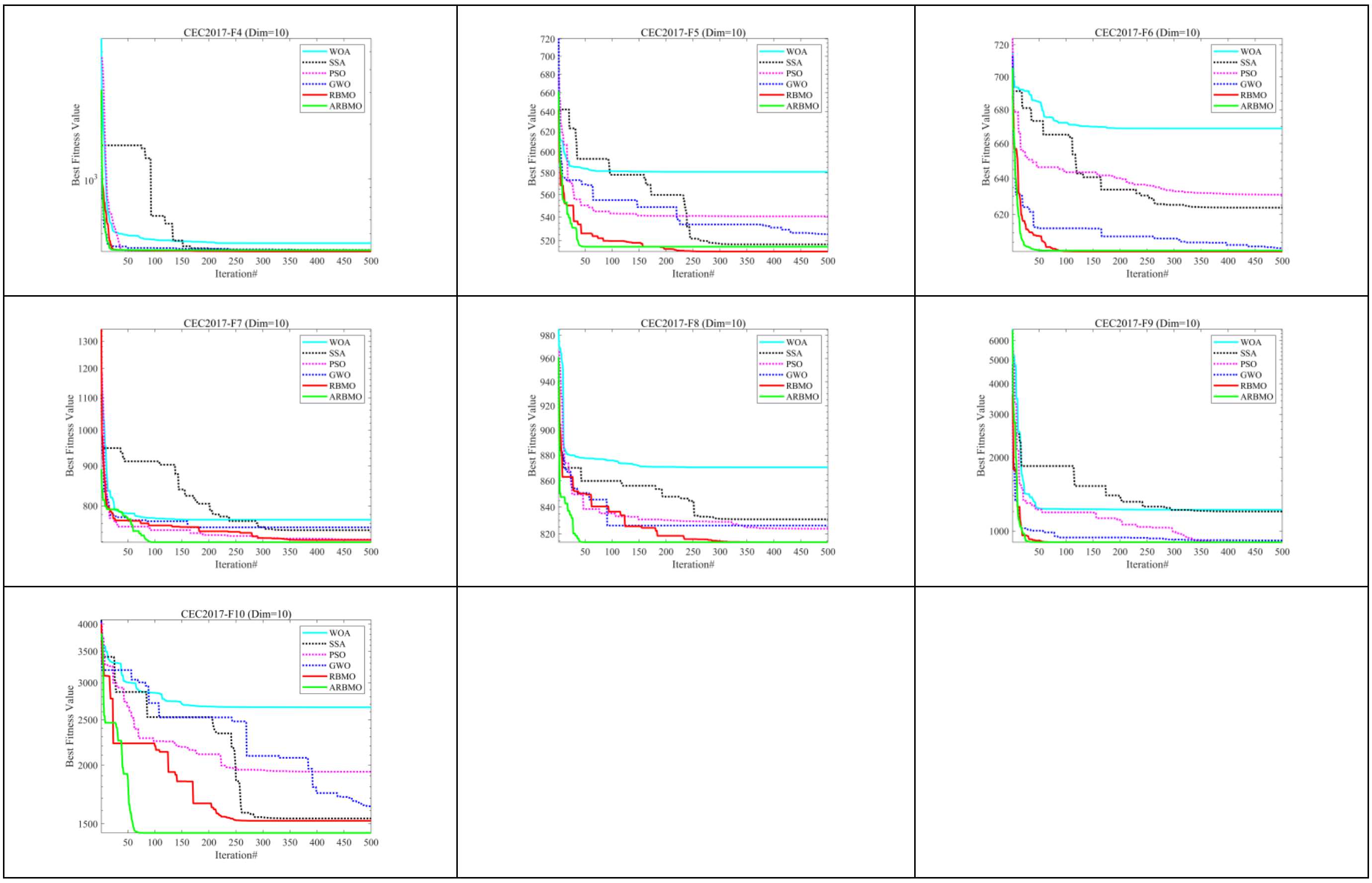


Fig. 2 Convergence curve of Simple multimodel functions

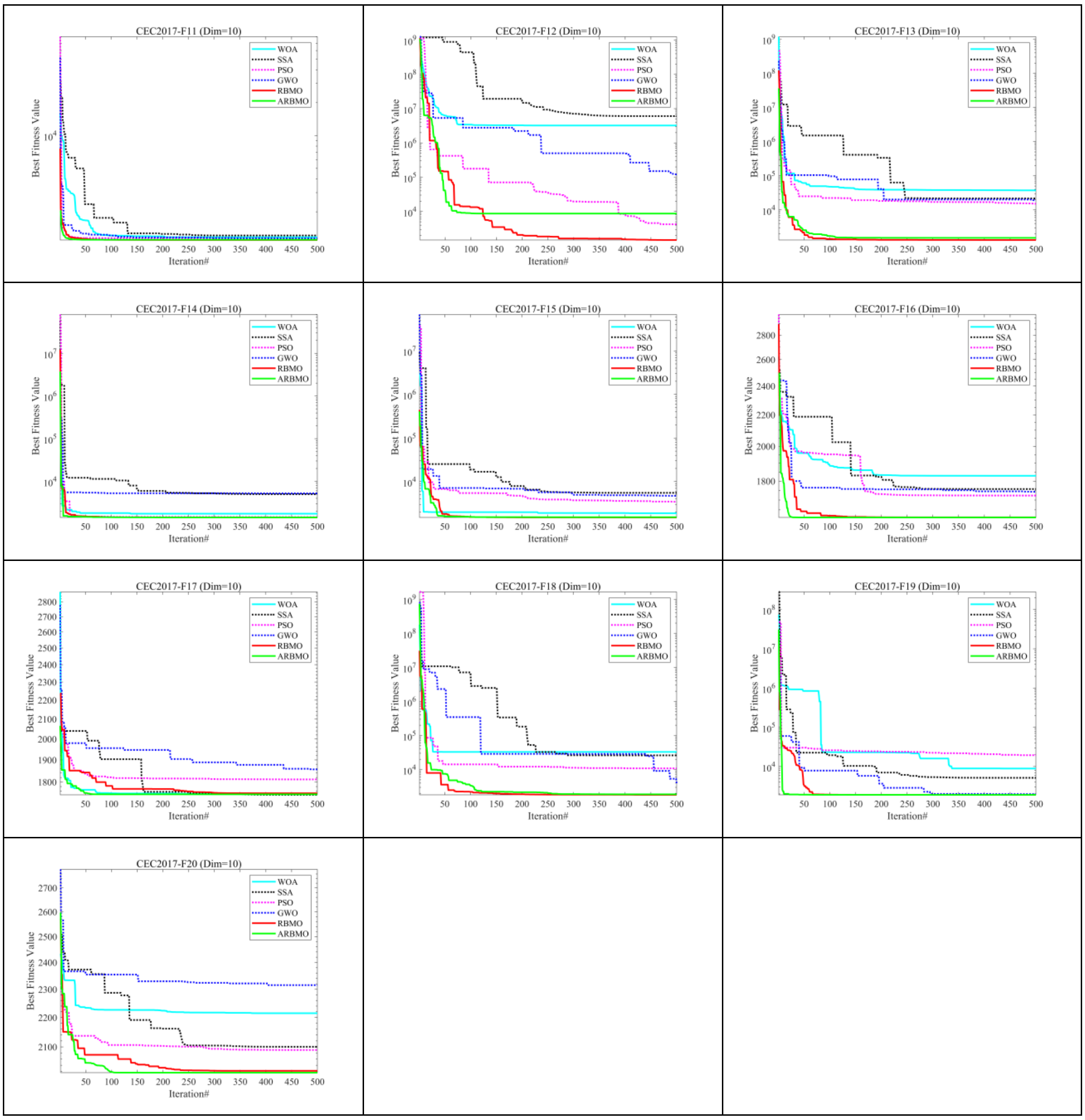
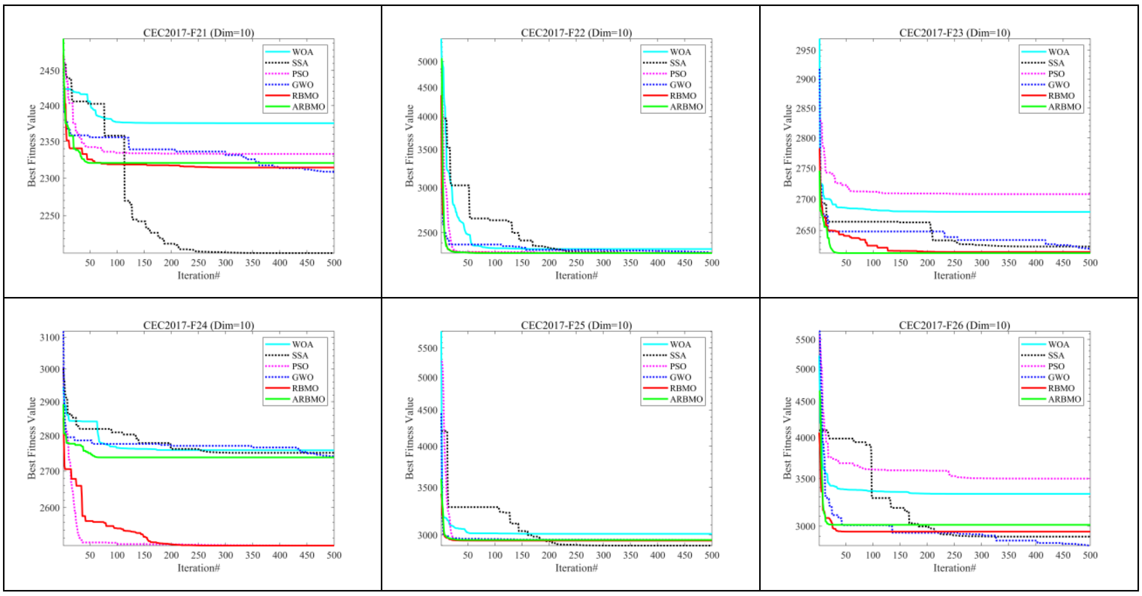


Fig. 3 Convergence curve of Hybrid functions



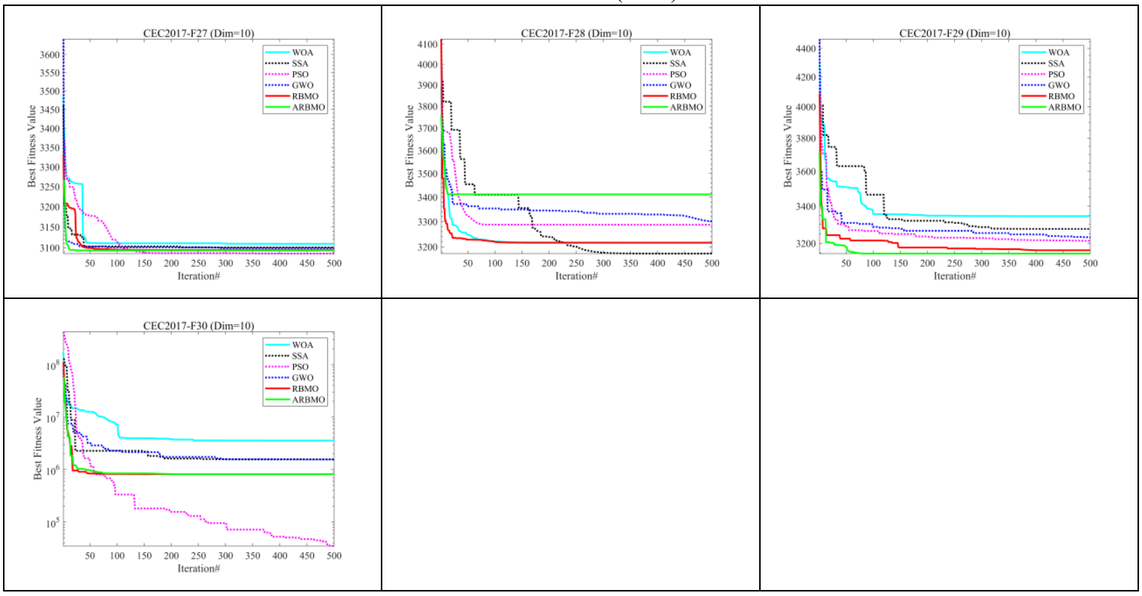


Fig. 4 Convergence curve of Composition functions

For the scheduling problem of the electric vehicle stator production line, aiming at minimizing the total inventory and the cumulative start date, and considering constraints such as orders, production takt time, shift working hours, minimum and maximum storage levels, and inventory, a weighted single-objective integer linear programming model is established for the scheduling of the electric vehicle stator production line.

$$min\, f = \sum_i \sum_j c_{ij} + \sum_i y_i$$

$$s.t.\begin{cases} x_{ij} > n_{ij} - c_{i-1j} \\ c_{ij} + n_{ij} = x_{ij} + c_{i-1j} \\ c_{1j} + n_{1j} = x_{1j} + K_j \\ L_j \le c_{ij} \le U_j \\ \sum_{j=1}^{m} x_{ij} \le t \cdot y_i \cdot P_j \\ y_i = 0,1; x_{ij} \in Z^+ \end{cases}$$

Where, x_{ij} denotes the quantity of the i-th type of stator produced on the j-th day; y_i is a 0-1 variable indicating whether production occurs on the j-th day for the i-th type of stator; c_{ij} represents the inventory of the i-th type of stator on the j-th day; n_{ij} denotes the demand quantity for the i-th type of stator on the j-th day; K_j represents the initial inventory quantity of the j-th type of stator; L_j and U_j denote the lower and upper limits of the inventory for the i-th type of stator, respectively; P_j denotes the production time for the j-th type of stator; t represents the actual working time of the current production shift.

Acknowledgements

This research is supported by the Major Science and Technology Projects of Jilin Province and Changchun City (20220301031GX)

References

[1] Qiu Y, Yang X, Chen S. An improved gray wolf optimization algorithm solving to functional optimization and engineering design problems[J]. Scientific Reports, 2024, 14(1): 14190.

[2] Kennedy J, Eberhart R. Particle swarm optimization[C]//Proceedings of ICNN'95-international conference on neural networks. ieee, 1995, 4: 1942-1948.

[3] Mirjalili S, Lewis A. The whale optimization algorithm[J]. Advances in engineering software, 2016, 95: 51-67.

[4] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer[J]. Advances in engineering software, 2014, 69: 46-61.

[5] Steinbrunn M, Moerkotte G, Kemper A. Heuristic and randomized optimization for the join ordering problem[J]. The VLDB journal, 1997, 6: 191-208.

[6] Wolpert D H, Macready W G. No free lunch theorems for optimization[J]. IEEE transactions on evolutionary computation, 1997, 1(1): 67-82.

[7] Li Y, Zhu X, Liu J. An improved moth-flame optimization algorithm for engineering problems[J]. Symmetry, 2020, 12(8): 1234.

[8] Hayyolalam V, Kazem A A P. Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems[J]. Engineering Applications of Artificial Intelligence, 2020, 87: 103249.

[9] Yıldız B S, Pholdee N, Panagant N, et al. A novel chaotic Henry gas solubility optimization algorithm for solving real-world engineering problems[J]. Engineering with Computers, 2021: 1-13.

[10] Sadollah A, Bahreininejad A, Eskandar H, et al. Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems[J]. Applied Soft Computing, 2013, 13(5): 2592-2612.

[11] Fu S, Li K, Huang H, et al. Red-billed blue magpie optimizer: a novel metaheuristic algorithm for 2D/3D UAV path planning and engineering design problems[J]. Artificial Intelligence Review, 2024, 57(6): 1-89.

[12] Yue Zhuo, Xingguo Song, Zhongqing Cao, Sheng Jin. Improved Algorithm of Spatio-Temporal Action Localization Based on YOWO. International Journal of Applied Mathematics in Control Engineering, 2024,7:1-9

[13] Zhonghua Han, Yu Zhang, Yuanyuan Liu, Optimization Research of Production Scheduling in Hybrid Production Process Workshops. International Journal of Applied Mathematics in Control Engineering, 2024,7:56-64

[14] Tianyu Chen, Lin Wang, Jiakai Yang, et al. Optimization Design of Burner Based on Three-dimensional Simulation of Rotary Kiln Temperature Field. International Journal of Applied Mathematics in Control Engineering, 2024,7:80-88

Jingru Liu currently works at FAW Mould Manufacturing Co., Ltd. She received her bachelor's degree from Northeast Agricultural University in 2012. Her main research interests lie in the development of intelligent equipment and visual control systems. She serves as the head of the Technology Innovation Laboratory in the Strategy and R&D Center.

Lin Fu currently works at FAW Mould Manufacturing Co., Ltd. He received his bachelor's degree from Changchun University of Technology in 2013 and his master's degree from Jilin University in 2016. His main research interests are in the fields of intelligent precision manufacturing and the development of intelligent equipment. He serves as a research and development engineer at the Technology Innovation Laboratory in the Strategy and R&D Center.

Baoyan Zhao currently works at FAW Mould Manufacturing Co., Ltd. He received his bachelor's degree from Harbin Institute of Technology in 2008. His main research interests lie in the development of intelligent equipment and the manufacturing of automotive body parts. He serves as the General Manager of the Strategy and R&D Center.

Ming Ren received his bachelor's degree from Changchun University of Technology in 2021, and is currently enrolled in Changchun University of Technology pursuing his master's degree, with his primary research interests in big data analytics.

Siyi Liu received her bachelor's degree from Jilin Normal University in 2020 and is currently studying at Changchun University of Technology for a master's degree in science. her main research direction is big data analysis.