Enhanced Multi-objective Artificial Physics Optimization Algorithms for Solution Set Distributions

Yunpeng Zhang
Innovation and Entrepreneurship Center, North China Institute of Aerospace Engineering, Langfang 065000, China
E-mail: nciaezhang@163.com

Keywords: artificial physics optimization algorithm, multi-objective, solution set, elite learning, chaotic mutation

Received: August 29, 2024

To optimize the distribution of solution sets in multi-objective optimization algorithms, this study takes the artificial physics optimization algorithm as an example, and introduces the elite learning, inverse learning, bi-directional speed, and chaotic mutation strategies to guide the evolution and search of individuals. This study guides the algorithm to carry out multi-objective solving from three aspects: external archive set, optimal guided individual selection, and step size optimization. The experimental outcomes denote that the improved artificial physics optimization algorithm designed in this study has the best minimum convergence value, standard deviation, optimization times, and convergence performance on the test function. The population fitness curve performs well. At the same time, the super volume index is 0.958, and the solution running time on different test functions is 6.13s, 7.61s, 8.46s, 9.68s, and 10.77s, respectively, with the smallest value. The improved multi-objective artificial physics optimization algorithm achieves the lowest Generative Distance value of 0.126, the lowest Inverted Generative Distance value of 0.171, the lowest extensiveness evaluation value of 0.210, and the maximum distributiveness evaluation value of 0.989. The actual solution coverage is high. This study expands the solution ideas and methods for multi-objective optimization problems, significantly improving the performance of artificial physics optimization algorithms in solving multi-objective optimization problems.

Povzetek: Razvit je je izboljšan algoritem umetne fizikalne optimizacije z elitnim učenjem, kaotično mutacijo in dvosmerno hitrostjo za izboljšanje razporeditve rešitev v večkriterijskih optimizacijskih problemih, s poudarkom na stabilnosti in učinkovitosti.

1 Introduction

Multi-objective Optimization Problem (MOP) indicates the process of minimizing or maximizing multiple objective functions under given constraints. The objective functions of multiple decision vectors in MOP are usually contradictory to each other, and the goal of solving them is to make all objective functions as optimal as possible while meeting the design requirements [1]. MOP can achieve a balance between solving multiple objectives, improve the scientificity and rationality of related decision-making problems, and help decision-makers optimize resource allocation. In recent years, MOP has been applied in various fields through various optimization techniques, including energy scheduling planning, logistics optimization, and financial investment decision-making [2-3]. MOP has multiple optimal solutions, and the solutions of Multi-Objective Optimization (MOO) algorithms should have good convergence, diversity, non-dominance, and optimal distribution and stability. The diversity and distribution of the solution set are key indicators for evaluating the advantages and disadvantages of MOO algorithms [4]. The diversity of solution sets determines whether algorithms can explore the solution space more comprehensively and improve the quality of solution sets. The distribution of the solution set is mainly reflected in two aspects: uniformity and universality, reflecting the characteristics of the Pareto frontier algorithm. To maintain the distribution of the solution set, existing solutions include techniques such as fitness sharing, crowding distance, and non-dominated sorting, which reduce the fitness of similar solutions and calculate crowding levels to maintain the distribution of the solution set. However, it is mostly reflected in solving the problem of uniform distribution of the solution set, without considering the true and effective distribution of practical problems [5]. In the real world, there are distributional non-uniformity problems in the solution set, including inhomogeneous and discontinuous phenomena, but there are relatively few researches on solving the distributional non-uniformity problems of the solution set.

To address the issue of uniform and non-uniform distribution of MOP solution set, the Artificial Physics Optimization (APO) algorithm was selected as the research object, and optimization research was conducted on the distribution of MOP solution. Firstly, improved strategies such as elite learning, Opposition-based Learning (OBL), bidirectional velocity, and chaotic mutation were introduced for the APO algorithm. Then, multi-objective theory was introduced and uniform distribution of MOP solution set was achieved through various improvements. Finally, the optimization of non-uniform distribution problems was achieved through the use of maximum and minimum distances. This study innovatively used multiple improvement strategies to enhance the solving performance of the algorithm, providing new ideas and methods for MOP solving. At the same time, the application of very large and very small distances has achieved remarkable results in solving the problem of non-uniform distribution of the solution set, which fills the research gaps of the distributional non-uniform problems and optimizes the distributiveness of the solution set.

The research mainly consists of five parts. The first part is a review and summary of the current research status of MOP and solutions at home and abroad. The second part elaborates on the improvement of APO algorithm and the MOO process. The third part conducts solution testing and application analysis on the designed multi-objective APO algorithm. The fourth part compares the study with the results of the existing work and discusses the advantages of the improved strategy and the reasons for the existence of differences. The fifth part summarizes and generalizes the experimental results.

2 Related works

MOP is a mathematical problem in the field of multidimensional decision-making, which exists in various fields and has received extensive attention and research from numerous domestic and foreign scholars. To handle expensive MOP, Luo et al. decomposed MOP into different sub problems and assigned related sub problems to the same task group. At the same time, an MOO algorithm based on multi-task conditional neural processes was proposed, which learned the similarity between sub-problems through a joint agent model. This method could effectively measure and utilize useful similarity information, improving the accuracy and robustness of the algorithm [6]. To raise the optimization efficiency and convergence accuracy of aerodynamic shape optimization in high-dimensional design space, Cao et al. proposed a dual-layer parallel mixed algorithm framework for multi-objective and large-scale decision variables in aerodynamic shape optimization. This framework introduced multi-objective gradient operators, new population individuals, and achieved a balance between development and exploration through elite individual selection. The experiment findings indicated that the optimization efficiency of the algorithm was significantly improved [7]. Changes in environmental parameters could affect the Pareto front of dynamic MOP, making it difficult to obtain Pareto optimal solutions. To achieve an equilibrium between the quality of Non-dominated Solution (NDS) and the associated computational costs, Chen et al. put forth a hybrid dynamic multi-objective evolutionary optimization method driven by the environment, which designed selection criteria for optimization methods based on dynamic environmental characteristics and scheme switching costs. The experiment findings indicated that this method could balance the convergence and robustness of NDSs [8]. The increased dimensionality of decision variables and the sparsity of Pareto optimal solutions contributed to the elevated difficulty of the MOP. Gu et al. proposed a quadratic association vector and dynamic guidance operator search algorithm, which could use reference vectors to achieve quadratic association of individuals and use dynamic guidance operators to guide the population to evolve towards sparse optimal solutions. The simultaneous selection of individuals for cross-mutation resulted in enhanced convergence of the algorithm. The experiment demonstrated that the algorithm exhibited comprehensive performance [9].

Traditional multi-objective evolutionary algorithms lack maintenance of decision space diversity. Zhang et al. designed a multi-modal multi-objective evolutionary algorithm with independent evolutionary sub-problems. This algorithm utilized a two-stage environment selection strategy to ensure the convergence of the target space and the diversity of the decision space. The experiment findings indicated that the algorithm had strong performance competitiveness [10]. Zouache et al. designed a multi-objective Harris Hawk optimization algorithm with reinforced dominance relationships, which selected leaders' solutions from external archives to guide the population in search, and used reinforced dominance relationships to realize a balance between coverage and convergence of Pareto sets. The outcomes denoted that this method had better convergence performance [11]. To achieve high accuracy, diversity, and completeness of MOP, Cao et al. designed a reinforcement learning hyper heuristic scheme based on a multi-objective simulated annealing algorithm with reseeding. This method exhibited stronger performance compared to benchmark test cases and could be applied in structural damage and other fields [12]. The development of automatic text summarization systems is crucial, but there are still difficulties in extracting multiple document collections from specific domains. Abo-Bakr et al. regarded automatic text summarization as MOP and designed an evolutionary sparse multi-objective algorithm to optimize the summarization process. The experiment findings confirmed the effectiveness of this method [13]. MOO scheduling is an effective means to solve the problem of segmented ship painting. Bu et al. designed an artificial bee colony algorithm based on decomposition strategy for multi-objective solution. This algorithm integrated five neighborhood switching methods, ideal solution sorting techniques, solution swapping strategies, and two-stage encoding methods. The experiment findings indicated that this method could effectively solve the problem of segmented painting of ships [14].

More scholars have conducted research around the practical application of optimization algorithms. Liu S et al. extended the classical stochastic gradient method to a multi-objective optimisation problem and computed a stochastic multi-gradient direction by solving a quadratic subproblem [15]. Dan Z S et al. entered the non-dominated sorting genetic algorithm II of multi-objective optimization algorithms into the optimization of the structural design of a heat return heater for printed circuits, with the weighted sum of the inverse of the total heat transfer rate and the total pumping power consumption as the optimization objective [16]. Jiang S et al. optimized the traditional golden jackal optimization algorithm for the deficiencies of the traditional golden jackal optimization algorithm by introducing the concepts of sinusoidal-cosinusoidal, Cauchy variational, and tent-mapping inverse learning to enhance the global exploration and solving ability of the algorithm [17]. Malti A N et al. proposed a new hybrid optimization algorithm for multi-objective task scheduling problems in heterogeneous IaaS cloud environments, which introduces the crossover operator of evolutionary algorithms [18].

Table 1: Summary of related work.

Literatures	Algorithms	Improved strategies	Results	Limitations
Luo J et al. [6]	An MOO algorithm based on multi-task conditional neural processes	Multi-task learning, decomposing subproblems	The calculation of Gaussian process covariance matrix is avoided, which improves the accuracy and robustness of the model.	Only studied for computational scale optimization
Cao F et al. [7]	A two-tier parallel hybrid algorithmic framework combining multi-objective local search and global evolutionary mechanisms	Multi-objective gradient operator, new individuals, elite strategy	The multi-objective hybrid algorithm is less constrained by the number of dimensions; there is a significant improvement in optimization efficiency and convergence accuracy	Only studied for large-scale decision variables with expensive cost functions
Chen M et al. [8]	Environment-driven hybrid dynamic multi-objective evolutionary optimization approach	Optimizing selection criteria for multi-objective algorithms	Convergence and robustness of NDSs can be balanced	Studied for dynamic multi-objective optimisation problems
Gu Q et al. [9]	Quadratic association vector and dynamic bootstrap operator search algorithms	Reference vectors, dynamic bootstrap operator, improved cross mutation	The algorithm achieves the best performance on 66.7% of the problems tested	For large-scale sparse multi-objective optimization problems
Zhang J et al. [10]	Multi-modal multi-objective evolutionary algorithms based on independent evolutionary subproblems	k-nearest neighbour deletion strategy	The algorithm has competitive performance	Only improves the diversity of the algorithm's solution set distribution in the decision space
Zouache D et al. [11]	Multi-objective Harris Hawk optimization approach	External archive sets, reinforced dominance relations,	Better convergence performance	Doesn't improve solution set distributability in a targeted way
Cao P et al. [12]	Reinforcement learning hyper-heuristic scheme	Dominance, congestion distance, and hypervolume computation	Shows stronger performance compared to benchmark test cases	Completeness without considering distributivity
Abo-Bakr H et al. [13]	Evolutionary sparse multi-objective algorithms	Evolutionary strategies	Good application in automatic text summary extraction	Fewer improvement strategies

Bu H et al. [14]	Artificial bee colony algorithm based on decomposition strategy	Five neighbourhood switching methods, ideal solution ordering techniques, solution exchange strategies, and two-stage coding	Effective in solving the ship segment painting problem	Does not improve solution set distribution properties
Liu S et al. [15]	Stochastic multi-gradient method	quadratic sub-problems	The method can be applied to any stochastic multi-objective optimization problem.	Failure to improve the distributional properties of the solution set
Dan Z S et al. [16]	Un-dominated Sorting Genetic Algorithm II	/	The minimum deviation index obtained by the decision-making method is 0.076.	No algorithmic optimization
Jiang S et al. [17]	Golden Jackal optimization Algorithm	Sine-cosine, Cauchy variants, and tent-mapping inverse learning	The method performs well in terms of convergence speed and accuracy	Does not improve the distributional properties of the solution set
Malti A N et al. [18]	Hybrid optimization algorithm	Crossover operators	The results obtained confirm the advantages of the newly designed hybrid algorithm	Fewer improvement strategies

In summary, a summary of related research work is shown in Table 1. As seen in Table 1, there have been many studies on the solution and application of MOP, but most of them focus on the application, convergence efficiency, and solution quality of MOP. There is relatively little research on the distribution of solution sets. Therefore, research is conducted on the optimization of APO algorithm and multi-objective solution set distribution.

3 Distribution of multi-objective optimization solution set based on APO algorithm

Regarding the distribution of solution sets in multi-objective APO algorithm, the study first introduces improvement strategies to optimize the efficacy of APO algorithm. Then, grounded on the multi-objective concept, a novel MOO APO (MOAPO) algorithm is proposed.

3.1 Design of improvement strategy for APO algorithm

Firstly, the study launches an optimization design for the APO algorithm. The APO algorithm is a global optimization algorithm that draws inspiration from artificial physics, which establishes the rules of interaction forces between individuals based on Newton's second law, driving individuals to continuously update their positions and perform optimization searches under the rules of interaction forces. APO completes searches based on group behavior and has strong parallel capabilities. By simulating the laws of motion in physics, strong global search can be achieved with a certain degree of flexibility. The mapping relationship between APO and optimization algorithms is shown in Figure 1.

In Figure 1, the APO algorithm considers the solution space in the optimization problem as a physical space, and the solution of the problem as particles in the physical space, which move under the action of forces. The physical space mainly contains fictional interaction forces and repulsion forces. Interaction forces drive particles to move towards a better solution, while repulsion forces prevent particles from falling into local optima. Firstly, it randomly generates a certain number of particles and assigns them initial positions and velocities. The set of positions is defined as $X_i(t)=(x_{i,1}(t),x_{i,2}(t),...,x_{i,n}(t))$, where t represents time. The velocity set is defined as $V_i(t)=(v_{i,1}(t),v_{i,2}(t),...,v_{i,n}(t))$, where both velocity and position are within a certain threshold range. It sets the objective function based on the research problem and calculates the fitness value f of each particle to retain the optimal individual. The calculation of individual quality $m_i^{(d)}$ is shown in equation (1).

$$m_i^{(d)}=e^{\frac{f(x_{best}^{(d)})-f(x_i^{(d)})}{f(x_{worst}^{(d)})-f(x_{best}^{(d)})}} \tag{1}$$

In equation (1), d represents individual grouping. According to equation (1), individual quality is inversely proportional to fitness value. The workflow of APO algorithm is shown in Figure 2.

In Figure 2, after completing the fitness evaluation, it needs to calculate the resultant force and update the individual's position and velocity. Firstly, based on the fitness value and positional relationship of the particles, the interaction and repulsion forces acting on each particle are determined. The calculation process of force $F_{ij,k}^{(d)}$ is shown in equation (2).

$$F_{ij,k}^{(d)}=\begin{cases} Gm_i^{(d)}m_j^{(d)}\left(-x_{i,k}^{(d)}+x_{j,k}^{(d)}\right), f\left(x_j^{(d)}\right)\le f\left(x_i^{(d)}\right) \\ Gm_i^{(d)}m_j^{(d)}\left(x_{i,k}^{(d)}-x_{j,k}^{(d)}\right), f\left(x_j^{(d)}\right)> f\left(x_i^{(d)}\right)\end{cases} \forall i\ne j, i\ne best \tag{2}$$

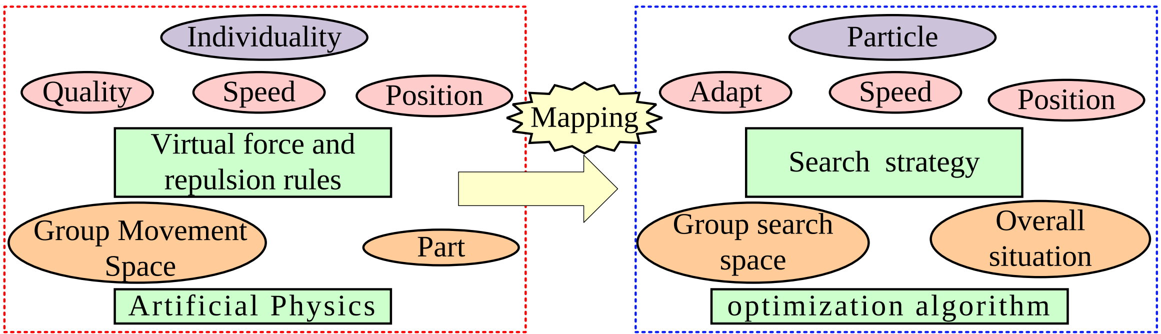


Figure 1: Schematic diagram of the mapping relationship between APO and optimization algorithm.

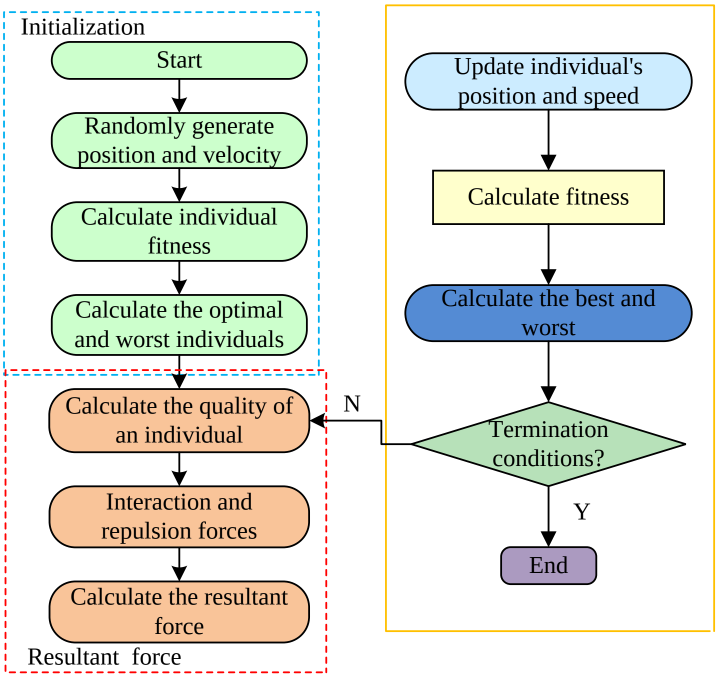


Figure 2: Schematic diagram of the workflow of APO algorithm.

In equation (2), G represents the gravitational factor, and G is within the range of [1, 100], and has a value of 10 based on prior empirical knowledge; $x_{i,k}^{(d)} - x_{j,k}^{(d)}$ represents the distance between individual i and individual j, $j, i \in m$. When individual i is greater than the fitness value of individual j, an interaction force occurs. k represents dimension. The calculation of resultant $F_{i,k}^{(d)}$ is shown in equation (3).

$$F_{i,k}^{(d)} = \sum_{j=1, i \neq j}^{m} F_{ij,k}^{(d)} \quad (3)$$

According to the force and motion rules of the particles, the position and velocity of the particles are updated. The calculation process is shown in equation (4).

$$\begin{cases} v_{i,k}(t+1) = wv_i(t) + \alpha F_{i,k} / m_i \\ x_{i,k}(t+1) = v_{i,k}(t+1) + x_{i,k}(t) \end{cases} \quad (4)$$

In equation (4), α represents a random variable, which takes values between (0, 1) according to prior empirical knowledge; w means the inertia weight, which determines the convergence effectiveness of the algorithm, while w is within the range of [0.1, 0.9]. However, traditional APO algorithms still have some shortcomings. Firstly, the randomly generated population samples during the initialization phase may exhibit a phenomenon of centralized distribution, which limits the search range of the algorithm and reduces its global search capability. Secondly, the APO algorithm has a single direction of individual resultant force, which facilitates the acceleration of individuals in the same direction during the iteration, resulting in limited search paths. During the final population iteration process, the diversity of the population continuously decreases, leading to the algorithm being prone to falling into local optima. To achieve better application results of APO algorithm in MOP, research is conducted on the quality improvement of APO algorithm.

The initial stage involves the introduction of an elite learning strategy, using elite individuals to guide the learning of other individuals, which serves to balance the global exploration capability and local optimization capacity of the algorithm [19-20]. The study divides elite learning into three phases: grouping, intra-group and inter-group learning. In the pre-grouping phase, each group needs to ensure that the region near the current better solution has been sufficiently searched to facilitate the subsequent mining of the better solution. As the evolution proceeds, the best individuals in each small neighbourhood are selected from each group to form elite individuals for information sharing and collaborative search. After completing the elite search, individuals from different groups conduct local fine search again around their respective elite individuals. The APO algorithm process based on elite learning strategy is denoted in Figure 3.

In Figure 3, the optimization process is composed of three stages. In the first stage, interval grouping strategy is adopted after randomly selecting the initial population individuals, dividing the search area equally according to the size of the coordinate values, with different areas containing the same individuals. Then, it calculates the fitness value of the individual and retains the dominant individuals. In the second stage, when the amount of iterations is less than the number of iterations within the group, intra-group learning is performed, and the resultant force of individuals is calculated to update their position and velocity, while retaining the dominant individuals. In the third stage, when the amount of iterations is greater than the amount of iterations within the group but less than the max amount of iterations, the top b optimal individuals outside the neighborhood r range are selected as elite individuals, while preserving the history and global optimality of the elite individuals. It randomly groups other individuals and searches around their respective elite individuals.

In addition, the study further introduces three improvement strategies based on elite learning, namely OBL, bidirectional velocity, and chaotic mutation, to enhance the population diversity of the algorithm and avoid it falling into local optima. To avoid the over-concentration of initial solutions in local regions caused by the random initial population generation approach, the study implemented the OBL strategy. OBL is designed based on the idea of exploration and balancing, which can generate a reverse solution based on a random initial solution during the population initialization process, ensuring a uniform distribution of the population, increasing the diversity of the search space, and helping the algorithm escape from local optima [21-22]. The OBL theory is shown in equation (5).

$$x_l^* = x_{l\min} + x_{l\max} - x_l \tag{5}$$

The bidirectional velocity is designed to prevent the APO algorithm from moving in a single direction, and should simultaneously consider the velocity or trend in two or more directions [23]. Dual-directional velocities originate from the idea of random wandering based on stochastic search algorithms, the individual's velocity direction is adjusted to achieve a wider search of the solution space, and the calculation process is shown in equation (6).

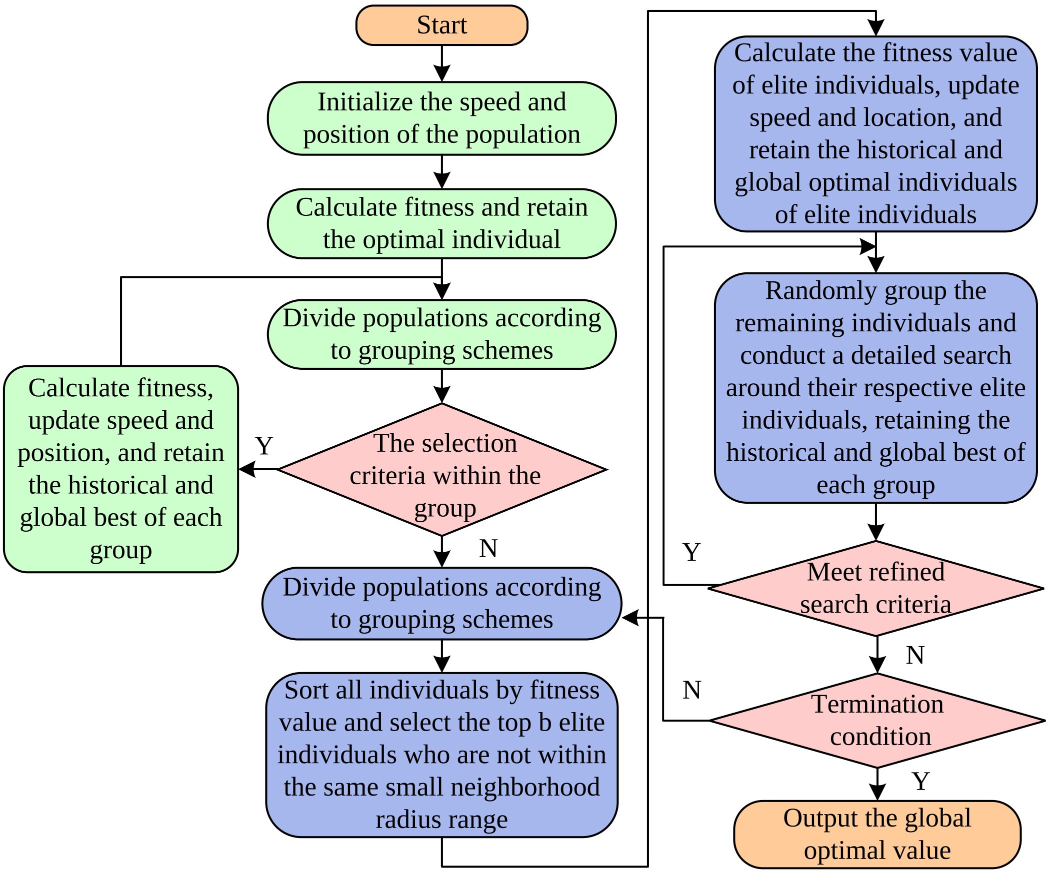


Figure 3: Schematic diagram of APO algorithm flow based on elite learning strategy.

$$x_{i,k}(t+1) = \begin{cases} x_{i,k}(t) + v_{i,k}(t), rand > \xi \\ x_{i,k}(t) - v_{i,k}(t), rand \le \xi \end{cases} \tag{6}$$

In equation (6), ξ represents the speed reverse walking control parameter, which takes the value of 0.2 based on prior empirical knowledge. After a certain amount of iterations of the APO algorithm, the population search ability decreases. The study introduces a chaotic mutation strategy to use the characteristics of chaotic variables to mutate individuals, increase population diversity, and help APO escape from local optima. The chaotic mapping β process is shown in equation (7) [24].

$$\beta = \frac{x_{best} - x_{\min}}{x_{\max} - x_{\min}} \tag{7}$$

Chaotic mapping variant operations are based on the properties of chaos theory, exploiting initial value sensitivity, long-term unpredictability and ergodicity to produce rich search behaviour. The chaotic variable is reversely mapped back to the solution space to obtain the position $x_{best}^{'}$ of the mutated point, as shown in equation (8).

$$x_{best}^{'} = x_{\min} + \mu\beta(1-\beta)(x_{\max} - x_{\min}) \quad (8)$$

In equation (8), μ represents a variable parameter used to control the chaotic state of the system and takes the value 4 based on prior empirical knowledge.

3.2 Design of multi-objective improved APO algorithm

After completing the improvement and optimization of the APO algorithm, to further solve the MOP problem, an MOPO algorithm based on the improved APO algorithm is proposed by combining MOO ideas. Moreover, the problem of solution set distribution is discussed from two aspects: uniform distribution and non-uniform distribution. The definition of multi-objective problem is shown in equation (9).

$$\begin{cases} \min y = F(x) = \left(f_1(x), f_2(x), \ldots, f_m(x)\right) \\ s.t. \begin{cases} g_i(x) \le 0, i = 1, 2, \ldots, q \\ h_j(x) = 0, j = 1, 2, \ldots, p \\ x \in [x_{\min}, x_{\max}] \end{cases} \end{cases} \quad (9)$$

In equation (9), x represents n decision variables in MOP; y represents m objective functions; $g_i(x)$ and $h_j(x)$ respectively represent q inequality constraints and p equality constraints. The goal of MOP solution is to find the Pareto front, which satisfies the premise that no other solution can improve any objective function without weakening other objective functions. The Pareto dominance relationship in the target space is shown in Figure 4.

In Figure 4, the dominance relationship reflects the comparison of the advantages and disadvantages between different solutions, and the Pareto NDS is the solution that achieves the best balance between different objectives. Pareto Dominated Solutions (DS) are inferior to other solutions in multiple objectives. The research on constructing MOAPO algorithm mainly focuses on three aspects: external archive set, optimal guidance individual selection, and optimization of travel step size.

The external archive set is mainly used to store and maintain Pareto NDS found during the optimization. The insertion of solutions in the maintenance process of external archive sets should follow two principles. Firstly, if a new solution dominates certain solutions in the archive set, the meaning of the DS will be removed and a new solution will be inserted. Secondly, it is notable that the novel solution, in conjunction with all other solutions within the archive set, does not exert dominance over the others, and the archive set has not reached its maximum capacity, allowing the insertion of new solutions. But when the archive set reaches its maximum capacity, other strategies need to be used to remove a solution from the archive set. To avoid premature convergence of the archive set, research is being conducted using adaptive grid method to maintain the diversity of the archive set. The target space is divided into several small spaces, and the calculation process of the mesh's modulus $F_i^{t'}$ is shown in equation (10).

$$\Delta F_i^{t'} = \frac{\max F_i^{t'} - \min F_i^{t'}}{M} \quad (10)$$

In equation (10), M represents the number of grids. The definition of grid number is shown in equation (11).

$$Int\left(\frac{F_i^k - \min F_i^{t'}}{\Delta F_i^{t'}}\right) + 1, i = 1, 2, \ldots, n \quad (11)$$

In equation (11), F_i^k represents the objective function value of the k th individual. Finally, it obtains density information of different grids and adaptively adjusts the grids based on individual threshold settings in the archive set. When Pareto NDS is unevenly distributed in the solution space, individuals in sparse regions are more likely to explore a wider solution space and become the globally optimal individuals. However, the possibility of higher quality solutions still exists in dense areas has been ignored. Therefore, the study introduces a game update mechanism to select the optimal guiding individual, and the operational mechanism is shown in Figure 5.

In Figure 5, the game mechanism is set in the external archive set, with the individual who wins the game as the

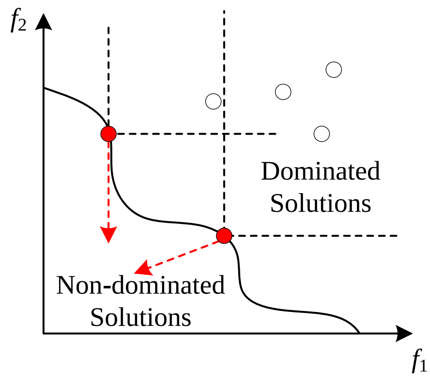


Figure 4: Schematic diagram of Pareto dominance relationship.

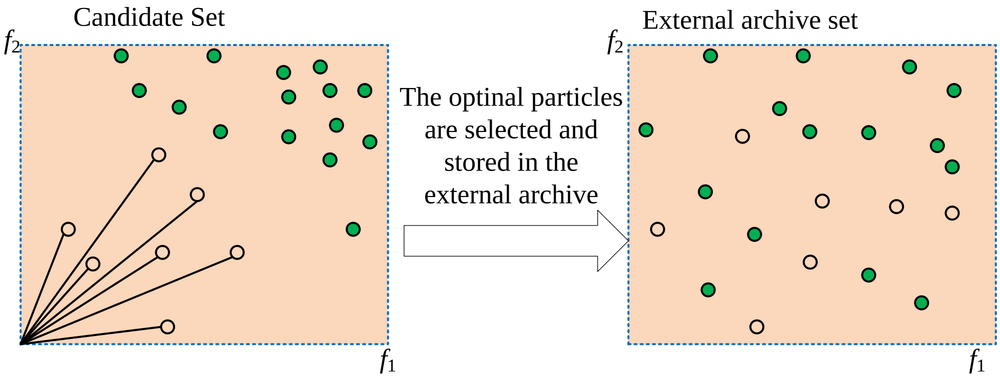


Figure 5: Schematic diagram of the optimal guidance individual game update mechanism.

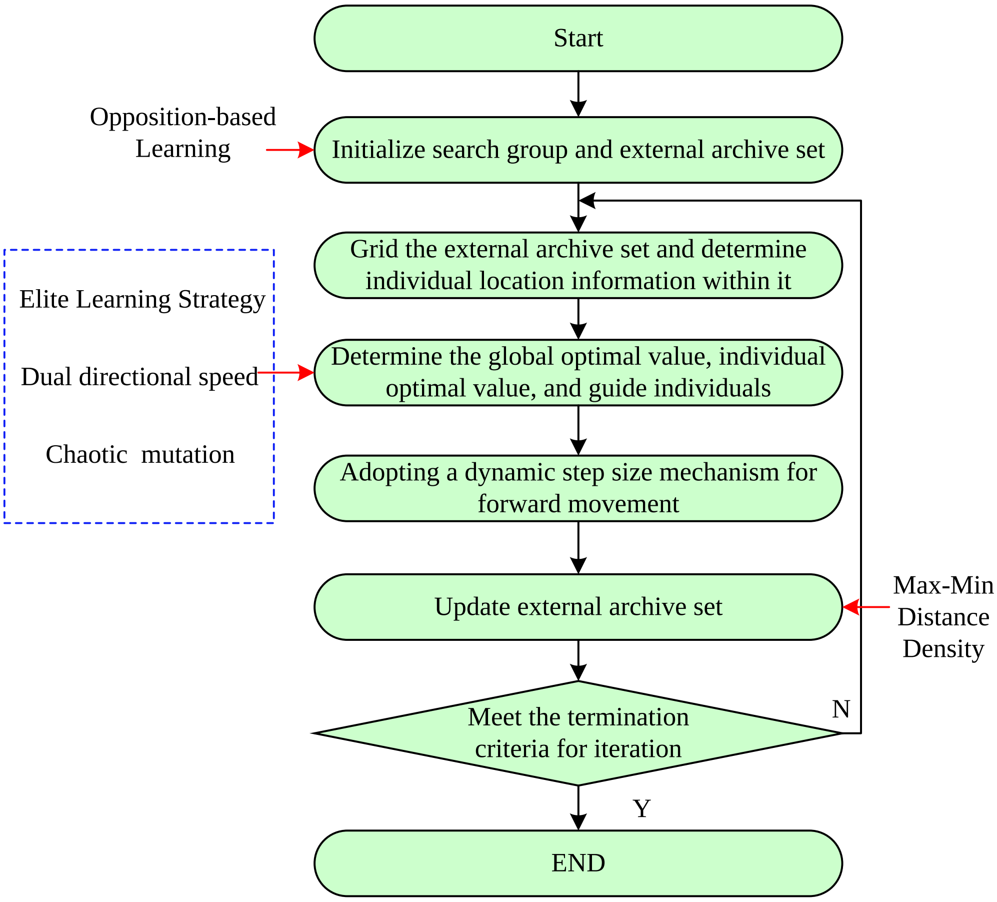


Figure 6: Schematic diagram of the workflow of the improved MOAPO algorithm.

guiding individual. The game updating mechanism can fully consider the interactions and influences between individuals, so that the algorithm can be more comprehensive and accurate in selecting the optimal guiding individuals. In addition, during the individual iteration process, optimizing the initial step size can effectively raise the accuracy of algorithm optimization. Therefore, the study introduces dynamic step size to raise the search speed and accuracy of the algorithm. The dynamic step size mechanism is shown in equation (12).

$$\begin{cases} \gamma = \exp \dfrac{\max time - Current\ time}{\max time} \\ Step = \gamma * AS - Step \end{cases} \tag{12}$$

In equation (12), $AS - Step$ represents the original step size; γ represents the control factor of the dynamic step size mechanism. Finally, to address the issue of non-uniform distribution of the solution set, the MOAPO algorithm is improved by introducing Max-Mini Distance Density (MMDD), which can measure the density of an individual in the objective function space. The individual set is defined as S, with a scale of n'. The Euclidean

Table 2: Comparison of comprehensive convergence performance of different optimization algorithms.

Test function	Algorithm	Average convergence value	Standard deviation value	Number of successful optimization attempts	Iteration times
Sphere	GA	2.4499E-04	0.535	34	639
	PSO	3.9043E-06	0.558	30	735
	APO	3.6148E-07	0.320	42	604
	Improved APO	2.9186E-12	0.142	50	567
Quartic	GA	1.3971E-04	0.483	34	647
	PSO	3.3631E-05	0.297	39	610
	APO	1.9542E-06	0.286	44	561
	Improved APO	1.8003E-11	0.086	50	497
Schaffer	GA	3.0487E-05	0.536	36	772
	PSO	1.0685E-04	0.368	25	746

distance between any individual and other individuals is d_j^i, and the minimum Euclidean distance is $d_{\min}^i$; For all individuals in the set S, there exists a minimum distance set consisting of $d_{\min}^i$, where there exists a maximum value $d_{\max-\min}$, and $d_{\max-\min}$ denotes the max-mini distance of the set S. The definition formula for the max-mini distance density $D(i)$ is shown in equation (13).

$$D(i) = \sum_{j=1, j\neq i}^{n^i} \left(\operatorname{sgn}\left(d_{\max-\min} - d_j^i\right)\right) \qquad (13)$$

In equation (13), sgn represents a sign function. When the individual distribution is relatively sparse and the max-mini distances between individuals are large, introducing MMDD can improve the algorithm's ability to maintain solution diversity during the search. The workflow of the improved MOAPO algorithm is denoted in Figure 6.

In Figure 6, in the MMDD optimization algorithm section, the individual with the maximum $D(i)$ is placed in the external archive set, and the quality m_i calculation of the individual is shown in equation (14).

$$m_i = \exp\left(d_{i-\min} / d_{\max-\min}\right) \qquad (14)$$

In equation (14), $d_{i-\min}$ represents the distance between an individual and the individual with the highest $D(i)$. Then, it calculates the individual resultant force and updates the individual position and external archive set until the algorithm meets the stopping condition.

Taking the real problem solving as an example, it is assumed that there exists a two-dimensional planar point set containing an initial population of 100 random points, and different points correspond to the cost function value and the optimization objective value. The initialized external archive set is empty for storing NDSs; and all hyper-parameters are completed with a maximum of 1000 iterations. The cost function value and optimization objective value of all points are calculated, the points according to the Pareto dominance relation are sorted, and the NDSs are added to the external archive set. Then, it selects the bootstrap individuals with relatively small cost function values and optimization objective values from the external archive set, and according to the positional relationship between the bootstrap individuals and other individuals in the population, dynamically step the adjustment mechanism. Subsequently, it generates new solutions and calculates their cost function values and optimization objective values. They are compared with the solutions in the external archive set, the external archive set is updated to include all the NDSs, and the distribution of the solution set is adjusted by using MMDD to regulate the distribution of the solution set non-uniformity.

4 Performance testing of multi-objective optimization combined with APO algorithm

To verify the efficacy and solution quality of the improved APO algorithm and MOAPO algorithm designed in the research, algorithm performance testing experiments were conducted and the results were analyzed and discussed.

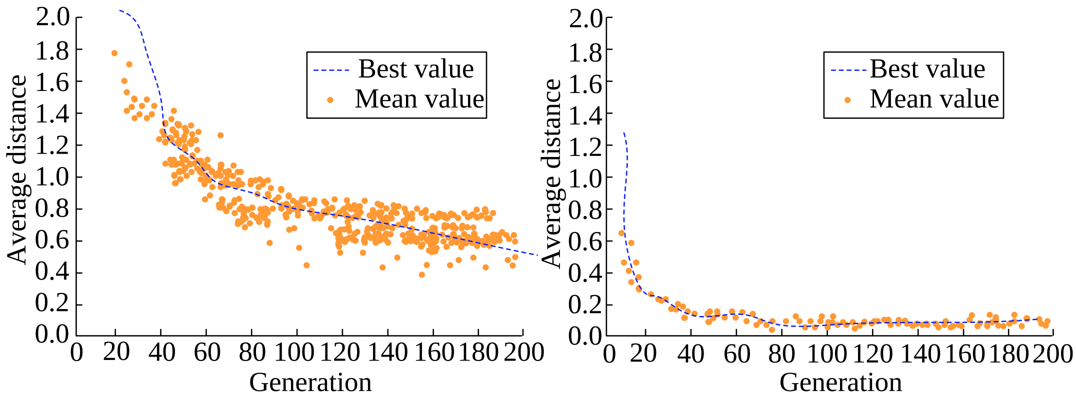
4.1 Performance testing of improved APO algorithm

The experiment was based on the Windows 10 operating system, with an Intel Core i7 central processor at 2.6 GHz and 128GB of memory. The image processor was Ge Force RTX 2080Ti. The experimental programming language was Python 3.8. Quantitative analysis of optimization algorithms were conducted using unimodal test functions Sphere and Quartic, as well as multimodal test functions Schaffer, Bohachevasky, and Eggrate. The Schaffer, Bohachevasky, and Eggrate functions had multiple local minima, which could test the effectiveness of the algorithm in exploring complex search spaces. Comparative algorithms included traditional APO algorithm, Particle Swarm Optimization (PSO) algorithm, and Genetic Algorithm (GA). The maximum number of iterations for the algorithm was set to 1000, the initial population size was 60, the dimension was 30, and the optimization times were 50. The neighborhood radius was 15 and the number of elite searches was 10. The comprehensive convergence performance comparison analysis outcomes are denoted in Table 2.

In Table 2, the improved APO algorithm performed better on the unimodal test function with a minimum convergence value of 2.9186E-12 on the Sphere function and 1.8003E-11 on the Quartic function. The convergence values of the other three algorithms were greater than those of the improved APO algorithm. The improved APO algorithm had an increased convergence mean on multimodal test functions, with Schaffer, Bohachevasky, and Eggrate functions of 4.0512E-09, 4.4789E-09, and 2.9270E-10, respectively. However, compared with other algorithms, it still achieved the best performance. The standard deviation of the improved APO algorithm was the smallest, all below 0.20. The maximum standard deviation value of PSO algorithm was 0.558. At the same time, the improved APO algorithm designed for research had a success rate of 100% in optimization, and the minimum number of iterations during convergence was around 500. Overall, the elite search and bidirectional speed strategy have improved the convergence and optimization ability of the algorithm, without any local convergence.

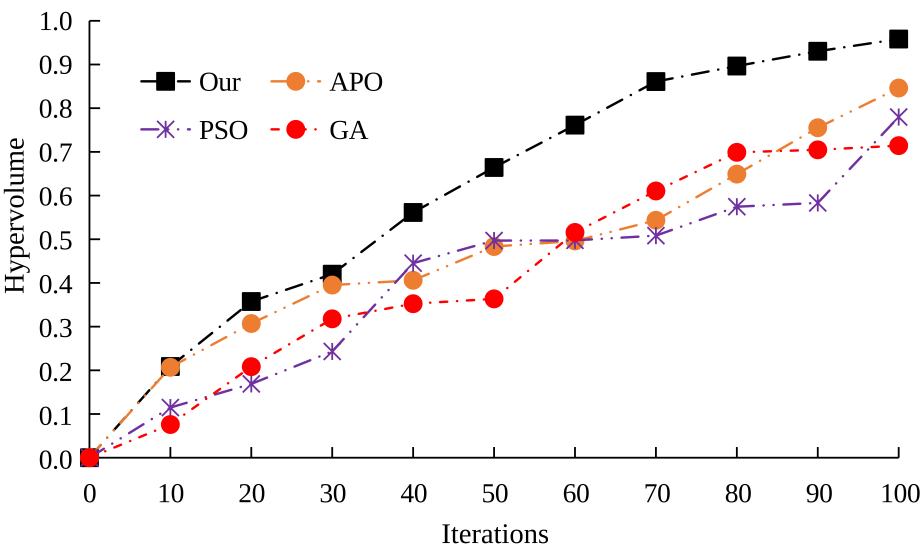
The optimization ability of the algorithm was evaluated before and after improvement, using the population fitness curve as the evaluation index. The experiment findings are indicated in Figure 7. In Figure 7 (a), the APO algorithm before improvement gradually approached the optimal population fitness in the later stages of iteration. In Figure 7 (b), in the early stage of iteration, the average population fitness of the improved APO algorithm approached the optimal population fitness. The population fitness ability of the improved APO algorithm has significantly improved, and the reverse learning and chaotic mutation strategies have improved the distribution and diversity of the initial population, thereby enhancing the algorithm's fitness ability.

Comparing the hypervolume (HV) index and solution running time of different optimization algorithms, the experiment outcomes are denoted in Figure 8. In Figure 8 (a), the HV curve of the improved APO algorithm was always at the highest level, and the growth rate of the entire iteration cycle was the fastest, with the maximum HV value reaching 0.958. By comparison, the maximum HV values of APO, PSO, and GA were 0.846, 0.779, and 0.714, respectively. HV measured the convergence and distribution of the solution set, and evaluated the algorithm by calculating the spatial HV enclosed by Pareto NDS and reference points. Experiment findings indicated that the improved APO algorithm had better convergence and distribution of the solution set than the other

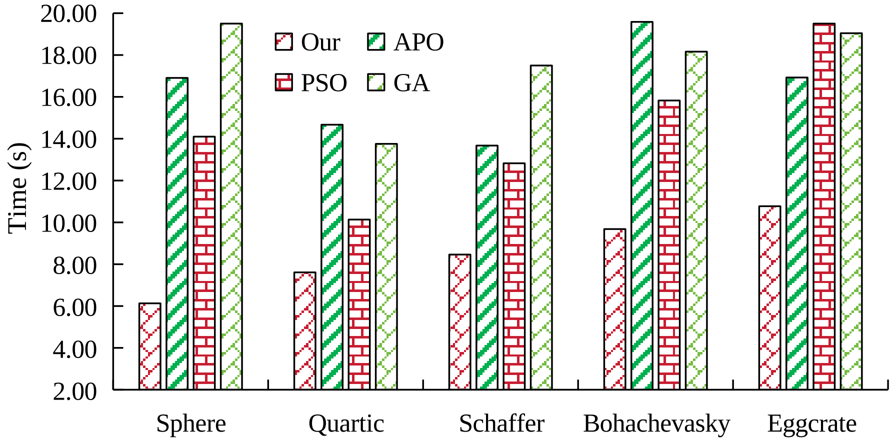


(a) Before improvement (b) After improvement

Figure 7: Comparison of population fitness curves before and after algorithm improvement.



(a) HV



(b) Times

Figure 8: Comparison of HV index and solution run time for different algorithms.

three algorithms. In Figure 8 (b), the computation running time on the five different test functions of the improved APO algorithm was the smallest, with values of 6.13s, 7.61s, 8.46s, 9.68s, and 10.77s, respectively. The longest computation times for APO, PSO, and GA were 19.58 s, 19.50 s, and 19.04 s, respectively. The improved APO algorithm had lower computational complexity and lower application costs. Usually, a higher HV value means that the algorithm is more capable of exploring the search space and can find more high-quality solutions, which will lead to an increase in the computational complexity of the algorithm and prolong the algorithm running time. However, the improved APO algorithm does not sacrifice the quality of the solution by pursuing computational efficiency. On the contrary, the PSO and GA algorithms perform poorly in the trade-off between HV value and computational efficiency.

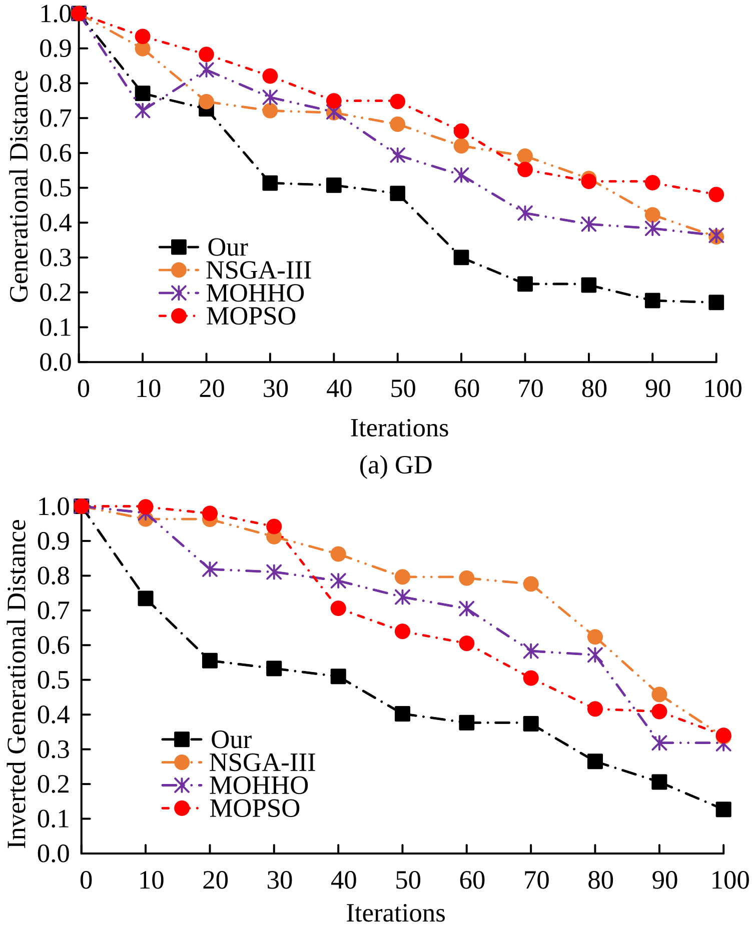
4.2 Performance testing of improved MOAPO algorithm

The improved MOAPO algorithm was contrasted with MOPSO, Non-dominated Sorting Genetic Algorithm III (NSGA-III), and Multi-Objective Harris Hawks Optimization (MOHHO) algorithm in reference [11]. Performance testing of MOO algorithm was conducted using SCH, ZDT3, ZDT6 dual objective function, and DTLZ1 triple objective function. The experimental results of Generative Distance (GD) and Inverse Generative Distance (IGD) for different algorithms are indicated in Figure 9. In Figure 9 (a), the GD curve of the improved MOAPO algorithm designed for research converged to the minimum value of 0.126, and remained at the lowest level after 20 iterations. By comparison, the minimum GD values of NSGA-III, MOHHO, and MOPSO algorithms were 0.336, 0.316, and 0.340, respectively. The GD value could evaluate the distance between the approximate optimal solution set and the true Pareto front. The GD value of the improved MOAPO algorithm was minimized, and the deviation between its solution result and the true optimal solution set was minimized. In Figure 9 (b), the IGD value of the improved MOAPO algorithm was also at its minimum, converging to 0.171. The minimum IGD values for NSGA-III, MOHHO, and MOPSO algorithms were 0.359, 0.364, and 0.480, respectively. The IGD value was the average Euclidean distance between the true Pareto front and the non-dominant solution, indicating that the distribution uniformity of the improved MOAPO algorithm has been improved.

The results of the Space Performance (SP) and Uniformity Performance (UP) evaluations for different multi-objective algorithms are denoted in Figure 10. In Figure 10 (a), the improved MOAPO algorithm had the lowest SP values on the four types of test functions. The minimum SP values on SCH, ZDT3, ZDT6, and DTLZ1 functions were 0.236, 0.233, 0.210, and 0.259, respectively. The SP values of NSGA-III and MOPSO algorithms were both above 0.23, with a maximum value of 0.666. SP represents the standard deviation of the minimum distance from different solutions to other solutions. The improved MOAPO algorithm had the best distribution. In Figure 10 (b), the UP value of the improved MOAPO algorithm could reach up to 0.989, and the uniformity of individual distribution in the algorithm solution set was relatively good.

The coverage metrics (C-metric) and Knee-driven dissimilarity (KD) results of different algorithms are shown in Figure 11. In Figure 11, the improved MOAPO algorithm designed for research had a higher value in the C-metric index, which was significantly different from other algorithms, with a maximum value of 0.845. The minimum C-metric value of MOPSO algorithm was 0.601. The design and solution of the research resulted in a wider distribution of the solution set in the target space, which could better cover and approach the real solution set. Meanwhile, the improved MOAPO algorithm had a smaller KD value, indicating that its solution set had a stronger ability to cover inflection points.

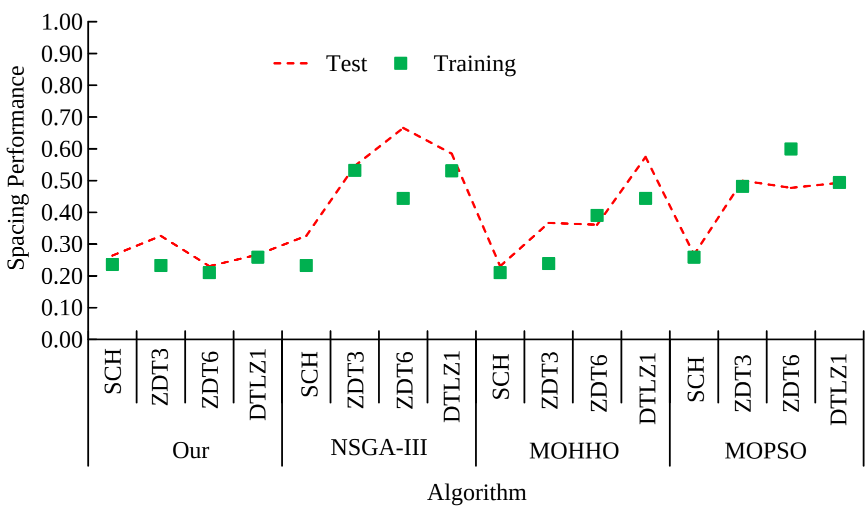
Finally, the cold chain logistics distribution path optimization taking into account the dynamic market demand response is the research object. The results of Friedman test and convergence comparison analysis of different methods are shown in Figure 12. In Figure 12(a), the improved MOAPO algorithm had the smallest number of iterations when the solution curve converged, and it reached the convergence level within 20 iterations, and the convergence value took the value of 0.02, which tended to be close to 0. Compared with other MOO algorithms, the convergence curve did not have the situation of local optimization for many times, and the global convergence was better. As seen in Figure 12(b), the improved MOAPO algorithm designed by the study obtained a rank-mean value of 3.75, which is ranked first. It can be seen that the design of the study achieved a more significant performance advantage.



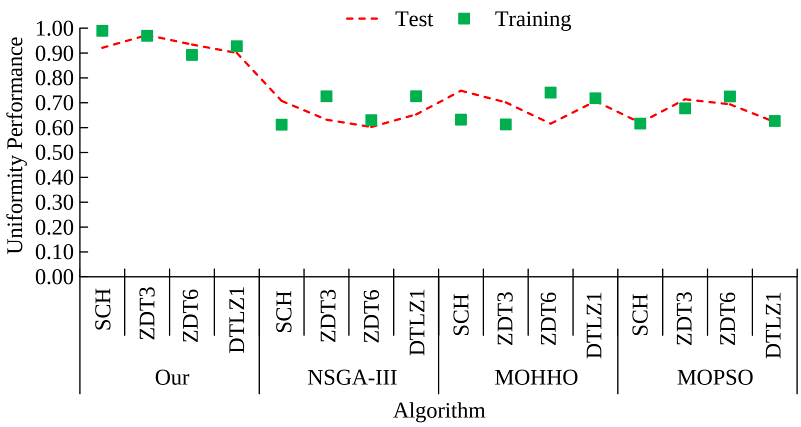
(a) GD

(b) IGD

Figure 9: Comparison of generative distance and inverse generative distance.



(a) SP



(b) UP

Figure 10: Comparison of spread performance and uniformity of different algorithms.

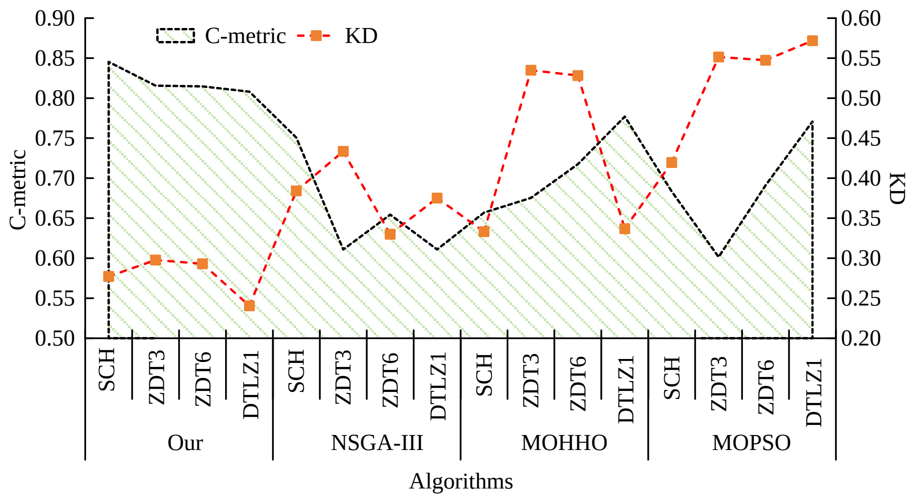


Figure 11: Coverage metrics and KD for different algorithms.

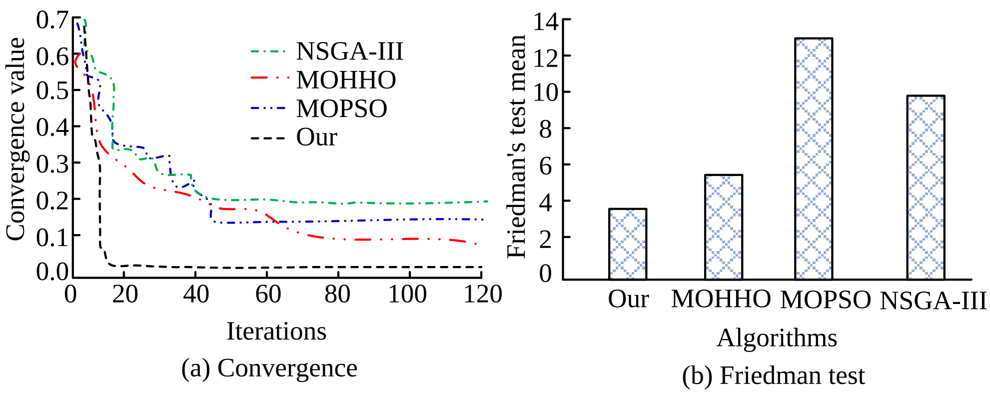


Figure 12: Comparative analysis of Friedman's test results and convergence.

5 Discussion

MOP designing multiple conflicting or interrelated objectives requires simultaneous optimization to find the superior equilibrium between multiple objectives. To solve MOP, researchers continue to propose a series of new algorithms, such as the multi-task conditional neural process-based multi-objective optimization algorithm proposed by Luo J et al. in reference [6], the environment-driven hybrid dynamic multi-objective evolutionary optimization method proposed by Chen M et al. in reference [8], and the quadratic correlation vectors and dynamic bootstrap operator search algorithms proposed by Gu Q et al. in reference [9]. These studies have enriched the variety of optimization algorithms and improved the optimization efficiency and convergence of the algorithms. However, in practical MOPs, such as path optimization, distribution network planning, power generation scheduling, etc., the distribution of the solution sets is not completely uniform, but the existing researches seldom focus on it. In contrast, the study took the APO algorithm as the object of research, which not only introduced elite learning, OBL, and dual vector speed to enhance the performance of the single-objective algorithm, but also extended the performance of the multi-objective optimization by combining the external archive set, the optimal bootstrap individual selection, and the travelling progress length optimization. Importantly, optimization for distributional non-uniform problems was achieved through very, very small distances. Compared to other research works, the improved MOAPO algorithm had richer and more targeted optimization measures, resulting in the lowest GD value of 0.126, the lowest IGD value of 0.171, the lowest SP value of 0.210, the maximum UP value of 0.989, and the best performance in terms of solution coverage. Combined convergence, distribution and diversity metrics, the improved MOAPO algorithm took better values than the MOHHO algorithm in reference [11].

6 Conclusion

MOP is an effective means of solving resource allocation, multi-objective decision-making, and system planning. To improve the non-uniformity of the solution set distribution of MOO algorithms, an improved design was carried out based on the APO algorithm. Moreover, multi-objective theory was introduced to optimize the distribution of the solution set. The experiment findings indicated that the minimum convergence value of the improved APO algorithm was 2.9186E-12, the standard deviation value was less than 0.2, and the optimization success rate of the improved APO algorithm was 100%. The algorithm had a maximum HV value of 0.958 and the shortest running time. The GD curve of the improved MOAPO algorithm converged to the minimum value of 0.126, and the IGD value converged to 0.171. The minimum SP values for the improved MOAPO algorithm on four types of test functions were 0.236, 0.233, 0.210, and 0.259, respectively. The UP value could reach up to 0.989. The C-metric index of this MOO algorithm was 0.845, and the KD value was relatively small. The improved MOAPO algorithm designed for research achieved better performance in the distribution of solution sets, promoting the advancement of MOP solving techniques.

However, there are still some potential drawbacks and challenges in the algorithm and its implementation. On the one hand, to maintain the diversity and quality of the solution set, the study introduced an external archive set to store the NDSs. However, the increase in the number of iterations and the increase in the size of the real-world problem both directly affected the size of the external archive set, which is related to the cost of storage and management. Memory requirements and operational efficiency are challenges to improve MOAPO in handling large-scale MOPs. On the other hand, the performance of the MOAPO algorithm relies heavily on a series of hyper-parameter settings. However, there is a lack of systematic and scientific theoretical guidance on how to make reasonable hyper-parameter settings according to the characteristics of actual problems. In future research work, research should continue to develop effective hyper-parameter tuning methods to improve the generality and robustness of the MOAPO algorithm. It also considers reducing the complexity of the algorithm while maintaining its performance, and promoting the application and development of the MOAPO algorithm in a wider range of fields.

References

[1] Zitong Wang, Yan Pei, and Jianqiang Li. A survey on search strategy of evolutionary multi-objective optimization algorithms. Applied Sciences, 13(7):4643-4667, 2023. https://doi.org/10.3390/app13074643

[2] Abhiroop Ghosh, Kalyanmoy Deb, Erik Goodman, and Ronald Averill. A user-guided innovization-based evolutionary algorithm framework for practical multi-objective optimization problems. Engineering Optimization, 55(12):2084-2096, 2023. https://doi.org/10.1080/0305215X.2022.2144275

[3] Siwakorn Anosri, Natee Panagant, Pakin Champasak, Sujin Bureerat, Chinnapat Thipyopas, Sumit Kumar, Nantiwat Pholdee, Betül Sultan Yıldız, and Ali Riza Yildiz. A comparative study of state-of-the-art metaheuristics for solving many-objective optimization problems of fixed wing unmanned aerial vehicle conceptual design. Archives of Computational Methods in Engineering, 30(6):3657-3671, 2023. https://doi.org/10.1007/s11831-023-09914-z

[4] Teeparthi Kiran, and Kumar D. M. Vinod. An improved artificial physics optimization algorithm approach for static power system security analysis. Journal of the Institution of Engineers (India): Series B: Electrical, Electronics & Telecommunication and Computer Engineering, 101(4):347-359, 2020. https://doi.org/10.1007/s40031-020-00457-6

[5] Hamid Shakibi, Navid Nedaei, Amir Hamzeh Farajollahi, and Ata Chitsaz. Exergoeconomic appraisement, sensitivity analysis, and multi-objective optimization of a solar-driven generation plant for yielding electricity and cooling load. Process Safety and Environmental Protection, 170(2):89-111, 2023. https://doi.org/10.1016/j.psep.2022.11.067

[6] Jianping Luo, Yongfei Dong, Qiqi Liu, Zexuan Zhu, Wenming Cao, Kay Chen Tan, and Yaochu Jin. A new multitask joint learning framework for expensive multi-objective optimization problems. IEEE Transactions on Emerging Topics in Computational Intelligence, 8(2):1894-1909, 2024. https://doi.org/10.1109/TETCI.2024.3359042

[7] Fan Cao, Zhili Tang, Caicheng Zhu, and, Xin Zhao. An efficient hybrid multi-objective optimization

method coupling global evolutionary and local gradient searches for solving aerodynamic optimization problems. Mathematics, 11(18):3844-3874, 2023. https://doi.org/10.3390/math11183844
[8] Meirong Chen, Yinan Guo, Yaochu Jin, Shengxiang Yang, Dunwei Gong, and Zekuan Yu. An environment-driven hybrid evolutionary algorithm for dynamic multi-objective optimization problems. Complex & Intelligent Systems, 9(1):659-675, 2023. https://doi.org/10.1007/s40747-022-00824-4
[9] Qinghua Gu, Yixiao Sun, Qian Wang, and Lu Chen. A quadratic association vector and dynamic guided operator search algorithm for large-scale sparse multi-objective optimization problem. Applied Intelligence, 53(16):19384-19405, 2023. https://doi.org/10.1007/s10489-023-04500-z
[10] Jialiang Zhang, Juan Zou, Shengxiang Yang, and Jinhua Zheng. An evolutionary algorithm based on independently evolving sub-problems for multimodal multi-objective optimization. Information Sciences, 619(1):908-929, 2023. https://doi.org/10.1016/j.ins.2022.10.096
[11] Djaafar Zouache, Adel Got, and Habiba Drias. An external archive guided Harris Hawks optimization using strengthened dominance relation for multi-objective optimization problems. Artificial Intelligence Review, 56(3):2607-2638, 2023. https://doi.org/10.1007/s10462-022-10235-z
[12] Pei Cao, Yang Zhang, Kai Zhou, and J. Tang. A reinforcement learning hyper-heuristic in multi-objective optimization with application to structural damage identification. Structural and Multidisciplinary Optimization, 66(1):16-35, 2023. https://doi.org/10.1007/s00158-022-03432-5
[13] Abo-Bakr H, Mohamed S A. Automatic multi-documents text summarization by a large-scale sparse multi-objective optimization algorithm. Complex & Intelligent Systems, 9(4):4629-4644, 2023.
[14] Henan Bu, Xianpeng Zhu, Zikang Ge, Teng Yang, Zhuwen Yan, and Yingxin Tang. A decomposition-based multi-objective evolutionary algorithm for solving low-carbon scheduling of ship segment painting. Coatings, 14(3):368-398, 2024. https://doi.org/10.3390/coatings14030368
[15] Liu Suyun, and Nunes Vicente Luis. The stochastic multi-gradient algorithm for multi-objective optimization and its application to supervised machine learning. Annals of Operations Research, 339(3):1119-1148, 2024. https://doi.org/10.48550/arXiv.1907.04472
[16] ZhiSong Dan, HuiJun Feng, LinGen Chen, NaiBing Liao, and YanLin Ge. Constructal design of printed circuit recuperator for S-CO2 cycle via multi-objective optimization algorithm. Science China Technological Sciences, 67(1):285-294, 2024. https://doi.org/10.1007/s11431-023-2500-x
[17] Shijie Jiang, Yinggao Yue, Changzu Chen, Yaodan Chen, and Li Cao. A multi-objective optimization problem solving method based on improved golden jackal optimization algorithm and its application. Biomimetics, 9(5):270-296, 2024. https://doi.org/10.3390/biomimetics9050270
[18] Arslan Nedhir Malti, Mourad Hakem, and Badr Benmammar. A new hybrid multi-objective optimization algorithm for task scheduling in cloud systems. Cluster Computing, 27(3):2525-2548, 2024. https://doi.org/10.1007/s10586-023-04099-3
[19] Weiwen Cai, Qihua Ma, Yazhe Wang, and Xuehui Gan. Crashworthiness analysis and multi-objective optimization of Al/CFRP hybrid tube with initial damage under transverse impact. Polymer Composites, 44(11):7953-7971, 2023. https://doi.org/10.1002/pc.27678
[20] Zhilin He. Improved genetic algorithm in multi-objective cargo logistics loading and distribution. Informatica, 47(2):253-260, 2023. https://doi.org/10.31449/inf.v47i2.3958
[21] Md Abu Talhamainuddin Ansary. A Newton-type proximal gradient method for nonlinear multi-objective optimization problems. Optimization Methods and Software, 38(3):570-590, 2023. https://doi.org/10.1080/10556788.2022.2157000
[22] Ali Nader Mahmed, and M. N. M. Kahar. Simulation for dynamic patients scheduling based on many objective optimization and coordinator. Informatica, 48(1):91-106, 2024. https://doi.org/10.31449/inf.v48i1.5256
[23] Matjaž Gams, and Tine Kolenik. Relations between electronics, artificial intelligence and information society through information society rules. Electronics, 10(4):514-529, 2021. https://doi.org/10.3390/electronics10040514
[24] Henrik Smedberg, and Sunith Bandaru. Interactive knowledge discovery and knowledge visualization for decision support in multi-objective optimization. European Journal of Operational Research, 306(3):1311-1329, 2023. https://doi.org/10.1016/j.ejor.2022.09.008