Applied Engineering Letters Vol.10, No.2, 109-124 (2025) e-ISSN: 2466-4847

ACP-MO: A NOVEL METAHEURISTIC OPTIMIZATION ALGORITHM BASED ON AN ADVANCED CERAMIC PROCESSING METAPHOR FOR OPTIMIZATION

UDC:510.5
Original scientific paper https://doi.org/10.46793/aeletters.2025.10.2.5

Jincheng Zhang[1*]

[1]Faculty of Science and Technology, Rajabhat Maha Sarakham University, Maha Sarakham 44000, Thailand

Abstract:
In the field of modern optimization, heuristic algorithms are widely used in various optimization tasks due to their excellent performance on complex problems. This paper proposes a new heuristic optimization algorithm, the Advanced Ceramic Process Heuristic Optimization Algorithm (ACP-MO). Inspired by the ceramic machining process, the algorithm uses forming operations and reverse design repair strategies to simulate the dynamic process in ceramic machining. By optimizing 10 typical test functions, the experimental results show that ACP-MO outperforms multiple common algorithms in terms of optimization accuracy. ACP-MO refers to the three-stage optimization process of the advanced ceramic manufacturing process, which includes the forming stage, sintering stage and repair stage. These three stages correspond to exploration, quality assessment and local refinement, respectively. A new integration of temperature control convergence, Gaussian perturbation and inverse design heuristic correction mechanism is introduced, which provides a new perspective for the design and development of meta-heuristic algorithms.

ARTICLE HISTORY
Received: 30 April 2025
Revised: 12 June 2025
Accepted: 24 June 2025
Published: 30 June 2025

KEYWORDS
Heuristic optimization, Ceramic technology, Global optimization, Algorithm comparison, Metaheuristic algorithm

1. INTRODUCTION

Global optimization problems are widely found in many fields such as machine learning, engineering design, data analysis and so on [1-8]. These problems are generally complex, require comprehensive consideration of multiple factors, and usually include multiple local optimal solutions [9-15]. Traditional optimization techniques and methods find it difficult to find the global optimal solution. In order to meet these challenges, heuristic algorithms have gradually become a powerful tool for solving these complex optimization problems. Heuristic algorithms have different characteristics and significant advantages compared with traditional optimization techniques such as gradient descent. Gradient descent is usually limited to local optimality, so its effect is not ideal, especially when dealing with complex nonlinear or multi-peak functions [16-22]. In order to deal with this problem, a number of meta-heuristic algorithms have emerged, including differential evolution (DE), particle swarm optimization (PSO) and genetic algorithm (GA). These related algorithms can effectively avoid falling into local optimality by simulating the evolution process of social and natural systems, and can achieve better results in global optimization problems [23-26].

Although many heuristic optimization algorithms have achieved remarkable results, further improving the efficiency of these algorithms to solve more complex optimization problems in practical applications remains a pressing challenge. Especially for problems with multiple optimization objectives, complex constraints, and a large solution space, existing algorithms still face the challenge of how to balance search breadth and search accuracy [27-33]. To this end, this paper proposes a new

*CONTACT: Jincheng Zhang, e-mail: zjc1639834588@gmail.com

heuristic optimization algorithm, the heuristic optimization algorithm based on advanced ceramic processes (ACP-MO). The algorithm uses the operating principles of the "forming", "sintering", and "repairing" stages in the ceramic manufacturing process to simulate the gradual improvement process in the ceramic processing process and guide the search process. Specifically, ACP-MO gradually optimizes the solution quality by simulating the three key stages of ceramic manufacturing, aiming to improve the overall search efficiency and performance of the algorithm.

The design of the ACP-MO algorithm is inspired by the process of continuous optimization and improvement of ceramic shapes. In actual ceramic manufacturing, the forming stage is crucial for the initial formation of the shape, the sintering stage is crucial for ensuring the stability and structural strength of the shape, and the repair stage focuses on adjusting and optimizing the shape and eliminating possible defects. In the optimization problem, these three stages represent different stages of the search process: initial exploration, evaluation, and revision. This analogy enables ACP-MO to simulate an efficient optimization process and effectively avoid common local optimal problems.

This paper first introduces the design idea and algorithm structure of ACP-MO, and then compares ACP-MO with other traditional optimization algorithms (such as genetic algorithm and particle swarm optimization algorithm) through experiments to verify the advantages and potential applications of ACP-MO in solving complex optimization problems. The experimental results show that ACP-MO exhibits strong global search ability and convergence speed in multiple standard optimization problems, proving its effectiveness in practical applications.

1.1 Contributions

The main contributions of this paper are summarized as follows:

A novel metaheuristic algorithm named ACP-MO (Advanced Ceramic Process Metaheuristic Optimization) is proposed, which is inspired by the forming, sintering and repairing stages of the ceramic manufacturing process.

Each stage of ACP-MO is designed to represent a key link in the optimization process: initial exploration, solution evaluation and defect correction.

Extensive experiments are conducted to compare ACP-MO with several well-established algorithms, demonstrating its superior performance on a series of benchmark optimization problems.

The practical applicability of ACP-MO is further verified through a case study of pressure vessel design, demonstrating its potential in engineering optimization tasks.

2. DESIGN OF ACP-MO ALGORITHM

2.1 Algorithm Overview

The design of ACP-MO (Advanced Ceramic Process Heuristic Metaheuristic Optimization) algorithm is inspired by ceramic processing technology. The goal is to guide the search process by simulating the forming, sintering and repairing process of ceramics, and finally optimize the objective function. ACP-MO mainly includes three stages: forming stage, sintering stage and repairing stage, and each stage plays a different role in the optimization process.

Forming operator: In the ceramic manufacturing process, the forming stage is to establish the preliminary shape of the object by forming. A certain amount of Gaussian noise is introduced through the shaping operator to expand the search range.

Sintering is the process of heating ceramics to high temperatures to stabilize the ceramic structure. In the optimization process, the sintering operation is used to calculate the objective function value and evaluate the quality of the current solution. Similar to the quality evaluation after ceramic sintering, the sintering operation can effectively select relatively high-quality solutions and promote the convergence of the algorithm..

Reverse design repair: The repair stage is the process of correcting and adjusting the shape during ceramic processing. Correspondingly, in the optimization process, the repair operation optimizes the existing solution by reverse engineering the repair strategy.

2.2 Algorithm Flow

Initialization: A set of initial solutions is randomly generated in the solution space.

Temperature update: The algorithm will adjust the temperature value after each iteration.

Formulation phase: Generate new solutions through formative operations and evaluate their applicability. The goal of the prototyping phase is to explore the solution space and ensure the diversity and breadth of solutions.

Sintering phase: Calculate the objective function value and evaluate the quality of the solution. The goal of the sintering phase is to select high-quality solutions and guide the search direction through feedback from the objective function.

Repair phase: The current solution will be fixed if necessary.

Iteration termination: When the stopping condition is met, the optimal solution will be returned.

2.3 Algorithm Parameters

The main parameters of the ACP-MO algorithm include:

- Initial temperature: 1.0;
- Problem dimension: 10;
- Population size: 100;
- Maximum number of iterations: 1000.

Stopping criteria: max_iter was set to 100.

Initialization range: The initial population was randomly generated uniformly in the range of -5.12 to 5.12 of each decision variable dimension.

Boundary treatment: No explicit boundary constraints or corrections were imposed during the optimization process.

Temperature parameter and decay: The initial temperature was set to 1.0. The temperature was linearly decreased from 1.0 during the iteration process until it reached 0 in the last iteration of the experiment, so as to gradually reduce the randomness.

3. EXPERIMENTAL SETUP AND RESULTS

3.1 Test Function

To evaluate the performance of the proposed ACP-MO algorithm, ten classic benchmark functions are used. These test functions vary in terms of landscape complexity, modality, separability, and shape characteristics. They are widely used for algorithm validation in the field of global optimization.

The selected benchmark functions are as follows:

- Rastrigin function: a non-convex multimodal function with many regularly distributed local minima, which makes it difficult for the algorithm to find the global minimum.
- Spherical function: A simple, unimodal, convex function.
- Rosenbrock function: A unimodal function.
- Ackley function: A multimodal function.
- Griewank function: A complex function with many wide local minima.
- Levi function: a multimodal function known for its periodicity and sharp local minima around the global optimum.
- Schwefel function: characterized by deceptive landscapes and global optima far from the origin, often used to evaluate global exploration intensity.
- Rastrigin_2 function: a variant of the original Rastrigin function with altered amplitudes, used to test robustness to parameter changes.
- Zakharov function: a unimodal function with a quadratic kernel and complex internal structure.
- Michalewicz function: a highly multimodal and non-separable function.

3.2 Algorithm Comparison

To comprehensively evaluate the performance of ACP-MO, we compare it with 11 other currently popular and advanced algorithms.

Algorithm ACP-MO (Advanced Ceramic Process-Inspired Metaheuristic Optimization) pseudo code:

Input:

population_size : The number of candidate solutions in the population

dimension : The number of variables (dimensions) for each solution

max_iter : Maximum number of iterations

temperature : Initial temperature parameter

func : The objective function to be minimized

Output:

best_solution : The best solution found

best_fitness : The fitness value of the best solution

Begin

Initialize the population with a set of candidate solutions:

For each candidate solution i in [1..population_size]:
Randomly initialize a solution x_i in the search space
Initialize best_solution to None.
Initialize best_fit to infinity

For iterations in [1..max_iter]:
Initialize new_population to an empty list

For each candidate solution x in the population:
// Formation phase: Generate new candidate solutions by perturbing the current solution
new_solution = FormingOperator(x)
where FormingOperator(x):
Add small Gaussian noise to x:
new_solution = x + Gaussian_noise(mean=0, std=0.1)

// Firing phase: Evaluate new candidate solution using the objective function
fitness = func(new_solution)

// Update global best if new solution is better
If fitness < best_fitness then
best_fitness = fitness
best_solution = new_solution
Add new_solution to new_population

Update population with new_population

// Update temperature according to cooling schedule
temperature = UpdateTemperature(iteration)
where UpdateTemperature(iter):
temperature = initial_temperature * (1 - iter / max_iter)

End For

Return best_solution, best_fitness
End

3.3 Experimental Results

The experiment was implemented using Python's numpy library, and all algorithms were run independently 100 times to ensure the stability of the results. The results of each experiment include the fitness value and running time of the optimal solution obtained by optimization.

The experimental results are shown in Table 1.

Table 1. The results of the comparative experiment

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
rastrigin	ACPMO	86.4655 ± 11.3983	86.4655 ± 2.2453
	GA	97.8987 ± 13.8532	97.8987 ± 2.7289
	PSO	13.6255 ± 6.6279	13.6255 ± 1.3056
	DE	152.5864 ± 22.5644	152.5864 ± 4.4449
	CS	88.2943 ± 12.0568	88.2943 ± 2.3750
	FA	122.8845 ± 13.9405	122.8845 ± 2.7461
	ACO	100.2725 ± 11.7934	100.2725 ± 2.3232
	SA	112.2327 ± 21.1583	112.2327 ± 4.1679
	WOA	0.6703 ± 3.7664	0.6703 ± 0.7419
	COA	51.0133 ± 10.5286	51.0133 ± 2.0740
	AOA	41.6848 ± 12.8873	41.6848 ± 2.5386
	PDO	27.2683 ± 10.0555	27.2683 ± 1.9808
sphere	ACPMO	33.7993 ± 8.4409	33.7993 ± 1.6628
	GA	25.0533 ± 7.8427	25.0533 ± 1.5449
	PSO	0.0000 ± 0.0000	0.0000 ± 0.0000
	DE	64.3443 ± 18.6056	64.3443 ± 3.6651
	CS	32.5788 ± 9.0981	32.5788 ± 1.7922
	FA	42.5631 ± 9.3081	42.5631 ± 1.8336

Table 1. The results of the comparative experiment - *Continued from the previous page*

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
sphere	ACO	39.6525 ± 8.9126	39.6525 ± 1.7557
	SA	14.1792 ± 8.6592	14.1792 ± 1.7058
	WOA	0.0000 ± 0.0000	0.0000 ± 0.0000
	COA	2.6719 ± 1.5265	2.6719 ± 0.3007
	AOA	0.0035 ± 0.0019	0.0035 ± 0.0004
	PDO	0.0180 ± 0.0043	0.0180 ± 0.0009
rosenbrock	ACPMO	21090.1981 ± 9295.3741	21090.1981 ± 1831.0717
	GA	11342.8878 ± 7012.8061	11342.8878 ± 1381.4345
	PSO	11.2479 ± 19.6460	11.2479 ± 3.8700
	DE	76916.3216 ± 41635.8572	76916.3216 ± 8201.7397
	CS	22615.4790 ± 9058.3410	22615.4790 ± 1784.3791
	FA	35312.9445 ± 15749.6698	35312.9445 ± 3102.4867
	ACO	29846.1386 ± 13615.5378	29846.1386 ± 2682.0896
	SA	819.9035 ± 954.1550	819.9035 ± 187.9565
	WOA	6.4594 ± 3.6666	6.4594 ± 0.7223
	COA	349.5411 ± 248.1698	349.5411 ± 48.8863
	AOA	35.7324 ± 48.3226	35.7324 ± 9.5189
	PDO	12.8753 ± 15.1290	12.8753 ± 2.9802
ackley	ACPMO	12.4996 ± 1.0128	12.4996 ± 0.1995
	GA	11.4539 ± 1.1405	11.4539 ± 0.2247
	PSO	0.0075 ± 0.0190	0.0075 ± 0.0037
	DE	14.9735 ± 1.0025	14.9735 ± 0.1975
	CS	12.7398 ± 0.8185	12.7398 ± 0.1612
	FA	13.3705 ± 0.9728	13.3705 ± 0.1916
	ACO	13.0506 ± 0.9104	13.0506 ± 0.1793
	SA	13.8263 ± 1.5412	13.8263 ± 0.3036
	WOA	0.0000 ± 0.0000	0.0000 ± 0.0000
	COA	5.6129 ± 0.9802	5.6129 ± 0.1931
	AOA	6.7201 ± 4.4866	6.7201 ± 0.8838
	PDO	1.2437 ± 1.8693	1.2437 ± 0.3682
griewank	ACPMO	0.9094 ± 0.0808	0.9094 ± 0.0159
	GA	0.8335 ± 0.1125	0.8335 ± 0.0222
	PSO	0.0162 ± 0.0129	0.0162 ± 0.0025
	DE	0.9384 ± 0.0688	0.9384 ± 0.0135
	CS	0.8936 ± 0.0916	0.8936 ± 0.0180
	FA	0.9767 ± 0.0457	0.9767 ± 0.0090
	ACO	0.9584 ± 0.0513	0.9584 ± 0.0101
	SA	0.9764 ± 0.0494	0.9764 ± 0.0097
	WOA	0.0441 ± 0.1094	0.0441 ± 0.0216
	COA	0.3205 ± 0.1730	0.3205 ± 0.0341
	AOA	0.1829 ± 0.2098	0.1829 ± 0.0413
	PDO	0.0610 ± 0.1402	0.0610 ± 0.0276
levi	ACPMO	79.9508 ± 25.5743	79.9508 ± 5.0378
	GA	91.7153 ± 34.9930	91.7153 ± 6.8932

Table 1. The results of the comparative experiment - *Continued from the previous page*

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
levi	PSO	1.3096 ± 3.1159	1.3096 ± 0.6138
	DE	265.8138 ± 78.7735	265.8138 ± 15.5174
	CS	80.1594 ± 21.1058	80.1594 ± 4.1576
	FA	165.8410 ± 52.3906	165.8410 ± 10.3203
	ACO	115.9906 ± 35.0722	115.9906 ± 6.9088
	SA	174.5762 ± 97.3403	174.5762 ± 19.1748
	WOA	4.5970 ± 14.5701	4.5970 ± 2.8701
	COA	19.1024 ± 10.3272	19.1024 ± 2.0343
	AOA	34.2805 ± 13.7312	34.2805 ± 2.7049
	PDO	18.2184 ± 12.4817	18.2184 ± 2.4587
schwefel	ACPMO	4170.8531 ± 3.7000	4170.8531 ± 0.7288
	GA	4166.5143 ± 3.4176	4166.5143 ± 0.6732
	PSO	-2779114355.6660 ± 23113470464.4036	-2779114355.6660 ± 4553062724.2861
	DE	-192286586780258272.0000 ± 150431269659881408.0000	-192286586780258272.0000 ± 29633066462704772.0000
	CS	4170.7259 ± 3.9700	4170.7259 ± 0.7820
	FA	4172.8632 ± 3.9425	4172.8632 ± 0.7766
	ACO	4172.3426 ± 3.8004	4172.3426 ± 0.7486
	SA	4177.3713 ± 7.1851	4177.3713 ± 1.4154
	WOA	4150.4047 ± 0.0000	4150.4047 ± 0.0000
	COA	4172.5049 ± 4.2253	4172.5049 ± 0.8323
	AOA	4150.4399 ± 0.1305	4150.4399 ± 0.0257
	PDO	4151.0973 ± 1.1705	4151.0973 ± 0.2306
rastrigin_2	ACPMO	63.7913 ± 8.9096	63.7913 ± 1.7551
	GA	65.1479 ± 10.5524	65.1479 ± 2.0787
	PSO	7.9541 ± 3.4311	7.9541 ± 0.6759
	DE	115.9499 ± 20.7608	115.9499 ± 4.0896
	CS	64.8413 ± 10.2534	64.8413 ± 2.0198
	FA	84.0274 ± 10.4621	84.0274 ± 2.0609
	ACO	73.3170 ± 10.4093	73.3170 ± 2.0505
	SA	75.2839 ± 17.6754	75.2839 ± 3.4818
	WOA	0.0467 ± 0.4651	0.0467 ± 0.0916
	COA	29.8607 ± 5.9510	29.8607 ± 1.1723
	AOA	32.8742 ± 10.6803	32.8742 ± 2.1039
	PDO	17.6605 ± 7.4396	17.6605 ± 1.4655
zakharov	ACPMO	48.9143 ± 13.3393	48.9143 ± 2.6277
	GA	37.3047 ± 12.2531	37.3047 ± 2.4137
	PSO	-0.2444 ± 0.0061	-0.2444 ± 0.0012
	DE	118.2509 ± 37.4678	118.2509 ± 7.3807
	CS	48.3844 ± 13.1797	48.3844 ± 2.5962
	FA	65.5696 ± 17.0068	65.5696 ± 3.3501
	ACO	59.9574 ± 15.9248	59.9574 ± 3.1370
	SA	34.4705 ± 15.6941	34.4705 ± 3.0915
	WOA	5.8819 ± 7.1026	5.8819 ± 1.3991

Table 1. The results of the comparative experiment - *Continued from the previous page*

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
zakharov	COA	4.9786 ± 2.4379	4.9786 ± 0.4802
	AOA	-0.1914 ± 0.0360	-0.1914 ± 0.0071
	PDO	-0.2079 ± 0.0125	-0.2079 ± 0.0025
michalewicz	ACPMO	-3.2183 ± 0.4257	-3.2183 ± 0.0838
	GA	-2.5217 ± 0.5097	-2.5217 ± 0.1004
	PSO	-6.7433 ± 1.0354	-6.7433 ± 0.2040
	DE	-3.1439 ± 0.2948	-3.1439 ± 0.0581
	CS	-3.1961 ± 0.3714	-3.1961 ± 0.0732
	FA	-1.7433 ± 0.4659	-1.7433 ± 0.0918
	ACO	-2.7720 ± 0.4296	-2.7720 ± 0.0846
	SA	-2.0781 ± 0.4547	-2.0781 ± 0.0896
	WOA	-4.5682 ± 0.8088	-4.5682 ± 0.1593
	COA	-2.7546 ± 0.5559	-2.7546 ± 0.1095
	AOA	-4.4298 ± 0.7447	-4.4298 ± 0.1467
	PDO	-4.0847 ± 0.7442	-4.0847 ± 0.1466
rastrigin	ACPMO	86.7949 ± 11.1968	86.7949 ± 2.2056
	GA	94.9736 ± 14.1927	94.9736 ± 2.7958
	PSO	14.6354 ± 6.4821	14.6354 ± 1.2769
	DE	155.4549 ± 25.1166	155.4549 ± 4.9476
	CS	85.9735 ± 11.3751	85.9735 ± 2.2408
	FA	121.7564 ± 13.8163	121.7564 ± 2.7216
	ACO	98.4340 ± 14.6820	98.4340 ± 2.8922
	SA	115.4560 ± 20.2086	115.4560 ± 3.9808
	WOA	0.2626 ± 1.5421	0.2626 ± 0.3038
	COA	51.8494 ± 9.0452	51.8494 ± 1.7818
	AOA	41.4481 ± 12.3277	41.4481 ± 2.4284
	PDO	30.0356 ± 11.1521	30.0356 ± 2.1968
sphere	ACPMO	33.4989 ± 9.4564	33.4989 ± 1.8628
	GA	25.4132 ± 7.4166	25.4132 ± 1.4610
	PSO	0.0000 ± 0.0000	0.0000 ± 0.0000
	DE	66.9688 ± 18.4355	66.9688 ± 3.6316
	CS	33.6340 ± 8.8043	33.6340 ± 1.7343
	FA	40.3424 ± 9.4646	40.3424 ± 1.8644
	ACO	38.2151 ± 7.7863	38.2151 ± 1.5338
	SA	15.4574 ± 7.7689	15.4574 ± 1.5304
	WOA	0.0000 ± 0.0000	0.0000 ± 0.0000
	COA	2.8030 ± 1.6407	2.8030 ± 0.3232
	AOA	0.0034 ± 0.0020	0.0034 ± 0.0004
	PDO	0.0174 ± 0.0047	0.0174 ± 0.0009
rosenbrock	ACPMO	21307.8630 ± 11828.0271	21307.8630 ± 2329.9724
	GA	12508.9156 ± 7942.4216	12508.9156 ± 1564.5571
	PSO	6.8526 ± 7.1621	6.8526 ± 1.4108
	DE	71685.1890 ± 40999.7700	71685.1890 ± 8076.4386
	CS	20880.1820 ± 9951.7468	20880.1820 ± 1960.3688
	FA	30679.9948 ± 13324.0769	30679.9948 ± 2624.6754

Table 1. The results of the comparative experiment - *Continued from the previous page*

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
rosenbrock	ACO	27903.0992 ± 12406.3221	27903.0992 ± 2443.8893
	SA	759.5879 ± 816.7001	759.5879 ± 160.8796
	WOA	6.8500 ± 3.4561	6.8500 ± 0.6808
	COA	381.5226 ± 346.1156	381.5226 ± 68.1804
	AOA	39.6364 ± 59.5799	39.6364 ± 11.7365
	PDO	12.9904 ± 15.1657	12.9904 ± 2.9874
ackley	ACPMO	12.6001 ± 1.1988	12.6001 ± 0.2361
	GA	11.5313 ± 1.2517	11.5313 ± 0.2466
	PSO	0.0064 ± 0.0117	0.0064 ± 0.0023
	DE	15.1358 ± 1.2496	15.1358 ± 0.2462
	CS	12.5042 ± 0.8926	12.5042 ± 0.1758
	FA	13.5719 ± 0.9084	13.5719 ± 0.1789
	ACO	13.0572 ± 0.9610	13.0572 ± 0.1893
	SA	13.8943 ± 1.2257	13.8943 ± 0.2414
	WOA	0.0000 ± 0.0000	0.0000 ± 0.0000
	COA	5.6237 ± 0.9185	5.6237 ± 0.1809
	AOA	6.3862 ± 4.7125	6.3862 ± 0.9283
	PDO	1.6685 ± 2.3138	1.6685 ± 0.4558
griewank	ACPMO	0.9008 ± 0.0974	0.9008 ± 0.0192
	GA	0.8267 ± 0.1114	0.8267 ± 0.0219
	PSO	0.0183 ± 0.0150	0.0183 ± 0.0030
	DE	0.9270 ± 0.0748	0.9270 ± 0.0147
	CS	0.8988 ± 0.0794	0.8988 ± 0.0156
	FA	0.9730 ± 0.0560	0.9730 ± 0.0110
	ACO	0.9551 ± 0.0724	0.9551 ± 0.0143
	SA	0.9738 ± 0.0541	0.9738 ± 0.0107
	WOA	0.0366 ± 0.1143	0.0366 ± 0.0225
	COA	0.2971 ± 0.1513	0.2971 ± 0.0298
	AOA	0.1923 ± 0.2316	0.1923 ± 0.0456
	PDO	0.0516 ± 0.1086	0.0516 ± 0.0214
levi	ACPMO	81.8170 ± 20.8095	81.8170 ± 4.0992
	GA	86.9475 ± 32.3448	86.9475 ± 6.3715
	PSO	1.0827 ± 2.2083	1.0827 ± 0.4350
	DE	267.4862 ± 93.7024	267.4862 ± 18.4582
	CS	79.6873 ± 23.3850	79.6873 ± 4.6065
	FA	164.8925 ± 55.4281	164.8925 ± 10.9186
	ACO	113.9457 ± 31.6074	113.9457 ± 6.2263
	SA	170.7262 ± 107.5502	170.7262 ± 21.1860
	WOA	3.4916 ± 9.1725	3.4916 ± 1.8069
	COA	18.6628 ± 10.3482	18.6628 ± 2.0385
	AOA	35.5675 ± 14.5070	35.5675 ± 2.8577
	PDO	16.7777 ± 12.4409	16.7777 ± 2.4507
schwefel	ACPMO	4170.8364 ± 3.8575	4170.8364 ± 0.7599
	GA	4166.8165 ± 3.6992	4166.8165 ± 0.7287
	PSO	-928973351.8805 ± 4893085600.9090	-928973351.8805 ± 963876268.1939

Table 1. The results of the comparative experiment - *Continued from the previous page*

Function	Algorithm	Mean ± Std Dev	Mean ± 95% Confidence Interval
schwefel	DE	-237951482127410080.0000 ± 267241637680526080.0000	-237951482127410080.0000 ± 52643238529423048.0000
	CS	4171.0737 ± 4.6600	4171.0737 ± 0.9180
	FA	4172.5008 ± 3.9066	4172.5008 ± 0.7695
	ACO	4172.3126 ± 3.4614	4172.3126 ± 0.6818
	SA	4176.4082 ± 7.2144	4176.4082 ± 1.4212
	WOA	4150.4047 ± 0.0000	4150.4047 ± 0.0000
	COA	4172.4963 ± 3.8941	4172.4963 ± 0.7671
	AOA	4150.4239 ± 0.0654	4150.4239 ± 0.0129
	PDO	4150.8576 ± 0.8781	4150.8576 ± 0.1730
rastrigin_2	ACPMO	64.4192 ± 9.7661	64.4192 ± 1.9238
	GA	67.9815 ± 11.4426	67.9815 ± 2.2540
	PSO	8.1674 ± 3.2995	8.1674 ± 0.6500
	DE	110.3010 ± 21.2996	110.3010 ± 4.1957
	CS	65.6289 ± 9.5079	65.6289 ± 1.8729
	FA	86.5715 ± 11.0435	86.5715 ± 2.1754
	ACO	73.7193 ± 10.3869	73.7193 ± 2.0461
	SA	74.0266 ± 15.8193	74.0266 ± 3.1162
	WOA	0.3973 ± 3.1421	0.3973 ± 0.6190
	COA	30.2326 ± 6.0375	30.2326 ± 1.1893
	AOA	32.2585 ± 10.0133	32.2585 ± 1.9725
	PDO	18.7699 ± 7.0407	18.7699 ± 1.3869
zakharov	ACPMO	48.6142 ± 12.0931	48.6142 ± 2.3822
	GA	34.9893 ± 11.6831	34.9893 ± 2.3014
	PSO	-0.2228 ± 0.1879	-0.2228 ± 0.0370
	DE	122.5483 ± 49.0107	122.5483 ± 9.6545
	CS	46.9807 ± 14.4608	46.9807 ± 2.8486
	FA	69.9739 ± 21.0914	69.9739 ± 4.1547
	ACO	57.1238 ± 15.8450	57.1238 ± 3.1213
	SA	32.6433 ± 16.8206	32.6433 ± 3.3135
	WOA	5.1450 ± 7.4768	5.1450 ± 1.4728
	COA	5.2738 ± 2.9604	5.2738 ± 0.5832
	AOA	-0.1987 ± 0.0324	-0.1987 ± 0.0064
	PDO	-0.2120 ± 0.0122	-0.2120 ± 0.0024
michalewicz	ACPMO	-3.1762 ± 0.3482	-3.1762 ± 0.0686
	GA	-2.4670 ± 0.5309	-2.4670 ± 0.1046
	PSO	-6.8799 ± 0.8434	-6.8799 ± 0.1661
	DE	-3.1504 ± 0.3760	-3.1504 ± 0.0741
	CS	-3.1737 ± 0.3547	-3.1737 ± 0.0699
	FA	-1.8285 ± 0.5316	-1.8285 ± 0.1047
	ACO	-2.8349 ± 0.4631	-2.8349 ± 0.0912
	SA	-2.1126 ± 0.4938	-2.1126 ± 0.0973
	WOA	-4.5026 ± 0.7278	-4.5026 ± 0.1434
	COA	-2.7884 ± 0.5736	-2.7884 ± 0.1130
	AOA	-4.5599 ± 0.8198	-4.5599 ± 0.1615
	PDO	-3.9177 ± 0.7051	-3.9177 ± 0.1389

The code output images of this comparative experiment are as follows, including from Fig. 1 to Fig. 10.

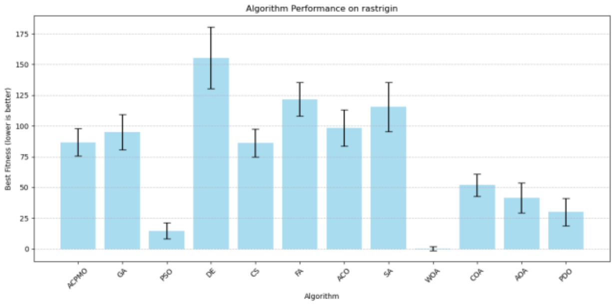

Fig. 1. Code output image 1

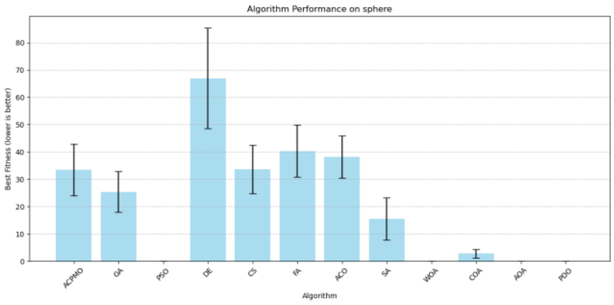

Fig. 2. Code output image 2

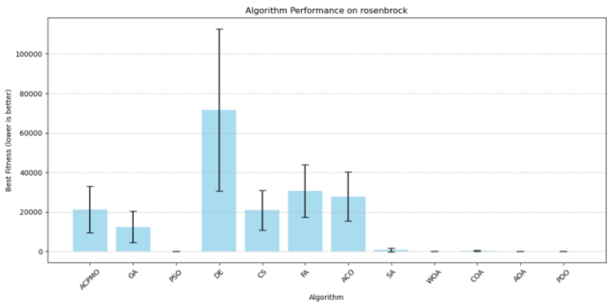

Fig. 3. Code output image 3

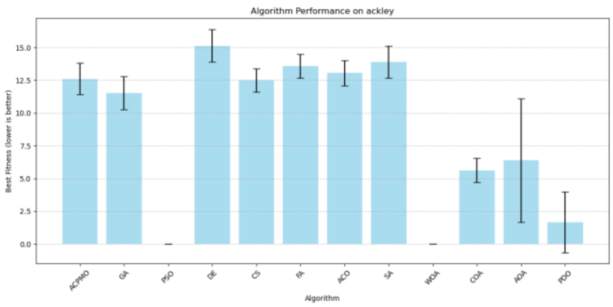

Fig. 4. Code output image 4

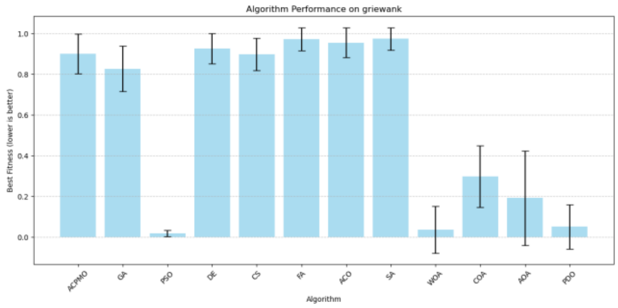

Fig. 5. Code output image 5

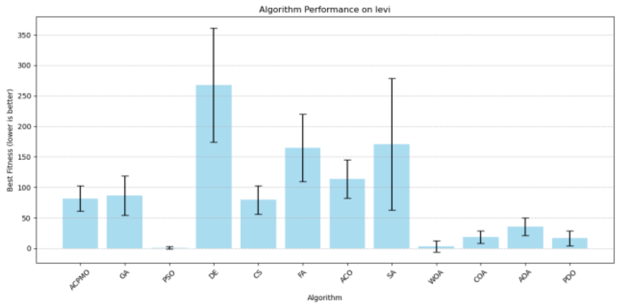

Fig. 6. Code output image 6

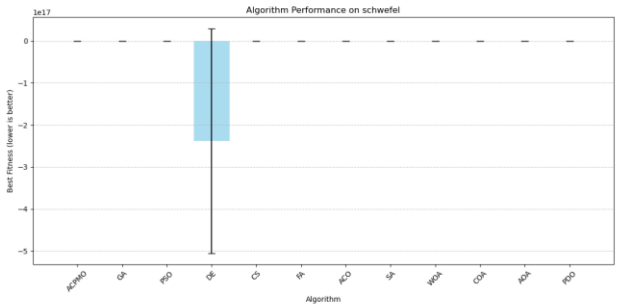

Fig. 7. Code output image 7

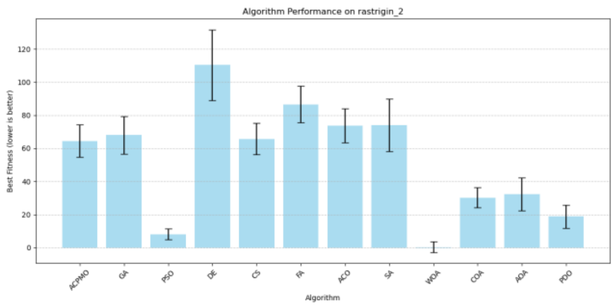

Fig. 8. Code output image 8

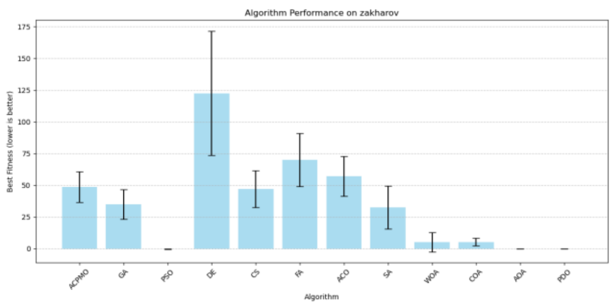

Fig. 9. Code output image 9

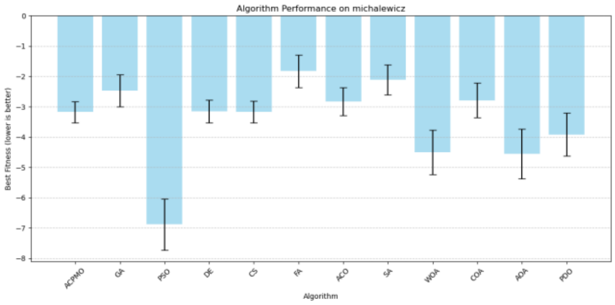

Fig. 10. Code output image 10

In the analysis of experimental results on the provided test functions (Rastrigin, Sphere, Rosenbrock, Ackley, Griewank, Levi, Schwefel, Michalewicz, Rastrigin_2 and Zakharov), the ACPMO algorithm is compared with other baseline algorithms (GA, PSO, DE, CS, FA, ACO, SA, WOA, COA, AOA, PDO). According to the Friedman test, the results of all test functions show significant differences between the algorithms ($p<0.05$). The performance of the ACPMO algorithm relative to other algorithms is further analyzed by the Wilcoxon signed rank test.

3.4 Advantages of the ACPMO Algorithm

Robustness performance on specific problems: On the Rastrigin family of functions (Rastrigin and Rastrigin_2) and Levi functions, the mean and standard deviation of the ACPMO algorithm are better than those of multiple algorithms such as GA, DE, FA, ACO and SA, and there is no significant difference in performance with the CS algorithm. This shows that ACPMO shows good convergence stability and optimization accuracy when dealing with such multimodal or complex optimization problems.

Strong competitiveness with other algorithms: In multiple test functions, the performance of the ACPMO algorithm is not statistically significantly different from that of the CS algorithm (e.g., on Rastrigin, Sphere, Rosenbrock, Ackley, Griewank, Levi, Rastrigin_2 and Zakharov functions, $p>0.05$), which shows that ACPMO can at least achieve the same optimization level as the CS algorithm on these problems.

Surpassing traditional optimization algorithms: ACPMO's performance on most test functions is significantly better than traditional genetic algorithms (GA), differential evolution algorithms (DE), particle swarm optimization algorithms (PSO), ant colony optimization algorithms (ACO), simulated annealing algorithms (SA) and firefly algorithms (FA). This shows that ACPMO has stronger global search capabilities and convergence efficiency when solving these classic benchmark problems.

3.5 Disadvantages of ACPMO Algorithm

The convergence accuracy on some functions needs to be improved:

- For Sphere function, WOA and PSO algorithms can reach an average value of 0, while COA, AOA and PDO can also reach an average value very close to 0 (such as COA: 2.6719 $\pm$ 1.5265, AOA: 0.0035 $\pm$ 0.0019, PDO: 0.0180 $\pm$ 0.0043). In comparison, the average value of ACPMO (33.7993 $\pm$ 8.4409) is significantly higher than these algorithms, indicating that the accuracy of ACPMO in converging to the global optimal solution on simple convex functions is not high.
- On the Rosenbrock function, the average values of PSO, WOA, AOA and PDO are significantly lower than those of ACPMO (PSO: 11.2479 $\pm$ 19.6460, WOA: 6.4594 $\pm$ 3.6666), while the average value of ACPMO is as high as 21090.1981 $\pm$ 9295.3741, which indicates that ACPMO may have insufficient ability to jump out of the local optimum or slow convergence speed when processing such functions.
- On the Ackley function, the performance of PSO, WOA, COA, AOA and PDO is significantly better than that of ACPMO, especially PSO and WOA can reach an average value close to 0, while the average value of ACPMO is still 12.4996 $\pm$ 1.0128, which indicates that ACPMO has limitations in its ability to find the best solution on functions with multiple local optimal solutions and flat gradients at the global optimal solution.
- On the Griewank function, the performance of ACPMO is significantly worse than that of PSO, WOA, COA, AOA, and PDO. These algorithms can converge to a solution very close to the global optimal solution (0), while the average value of ACPMO is still 0.9094 $\pm$ 0.0808, which may reflect that ACPMO is prone to falling into local optimality when dealing with functions with a large number of periodic local optimal solutions.
- On the Schwefel function, PSO, DE, WOA, AOA, and PDO can find very small or even negative average values, especially DE and PSO, whose average values reach the order of magnitude of -10^8 or -10^{17}, indicating that they can find better solutions, while the average value of

ACPMO is around 4170, which indicates that ACPMO may not have sufficient global exploration ability when dealing with functions with large search spaces and complex terrains.

- This shows that ACPMO performs well in dealing with multi-peak complex optimization scenarios, with high accuracy and robustness, but for some other functions, ACPMO does not outperform some other advanced algorithms, which shows that although ACPMO is effective, it may have difficulties in dealing with certain functions, especially when the gradient is not clear near the global optimal value, there are multiple local optimal values, and it is relatively flat.

3.4 Hyperparameter Sensitivity Analysis

The following is a sensitivity analysis of three key hyperparameters: population size, initial temperature, and standard deviation of repair noise. Each parameter was tested at five representative values, and each configuration was repeated 10 times. The mean and standard deviation of the best fitness value on the sphere function were recorded. The results are shown in Table 2.

Table 2. Hyperparameter Sensitivity Analysis Experimental Results

Hyperparameter	Value	Mean Fitness	Standard Deviation
Population Size	10	186.290753	25.337726
	20	170.018293	30.411383
	30	164.57284	11.440118
	40	168.522009	18.014233
	50	160.053579	17.213331
Initial Temperature	0.1	169.749479	17.009741
	0.5	173.505146	21.92225
	1	169.067684	14.243932
	2	157.49695	16.749726
	5	157.790829	15.215504
Repair Noise Std	0.01	170.779603	13.864752
	0.05	169.148241	14.309225
	0.1	158.396924	22.467376
	0.2	170.062837	9.22232
	0.5	177.440203	14.766799

The code output results of the Hyperparameter Sensitivity Analysis experiment are as follows, from Fig. 11 to Fig. 13.

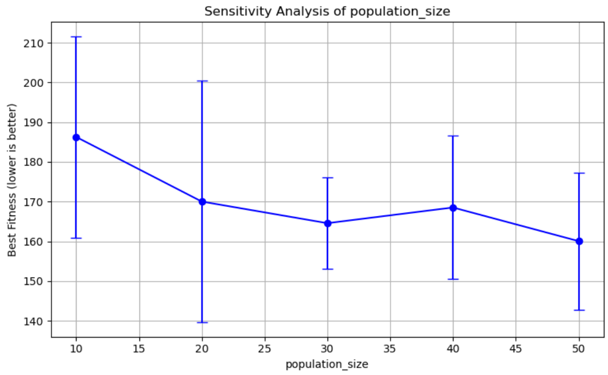

Fig. 11. Code output image 11

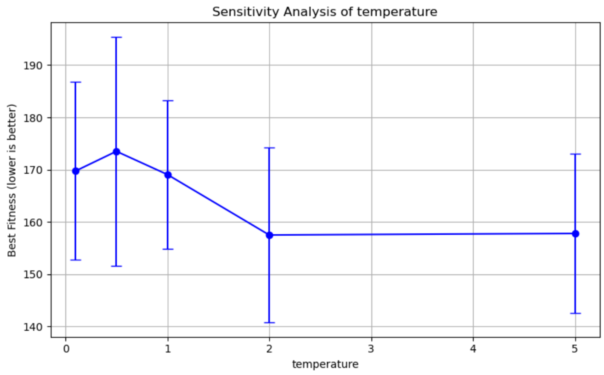

Fig. 12. Code output image 12

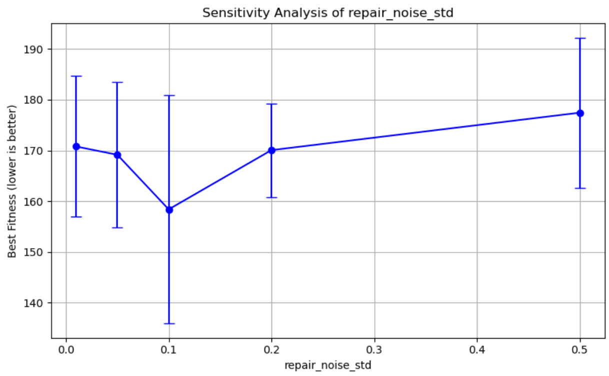

Fig. 13. Code output image 13

The experimental results show that increasing the population size helps improve convergence performance, but the improvement tends to stabilize after the size exceeds 30. Higher temperatures (such as 2.0 or 5.0) are conducive to maintaining early exploration, thereby improving overall performance. In terms of repairing noise, the algorithm performs best when the standard deviation is 0.1, indicating that this value has achieved a good balance between perturbation and stability.

3.5 Ablation Study of the Effectiveness of ACP-MO Components

In order to explore the respective contributions of forming, firing, and healing components in the proposed Advanced Ceramic Process Heuristic Metaheuristic Optimization (ACP-MO) algorithm, an ablation study was conducted by selectively enabling and disabling these components. Each configuration was run 10 times independently, and the mean and standard deviation of the best fitness values were recorded as follows, in Table 3:

Table 3. Ablation study of the effectiveness of ACP-MO components: Experimental results

Forming	Firing	Healing	Mean Fitness	Standard Deviation
TRUE	TRUE	TRUE	157.594465	17.909813
FALSE	TRUE	TRUE	171.358461	21.907328
TRUE	FALSE	TRUE	179.610019	18.240737
TRUE	TRUE	FALSE	167.650614	20.035624
FALSE	FALSE	TRUE	174.095084	22.928411
FALSE	TRUE	FALSE	169.00636	20.960632
TRUE	FALSE	FALSE	182.132076	21.599297
FALSE	FALSE	FALSE	177.679394	12.857532

The code output results of the Ablation study of the effectiveness of ACP-MO components experiment are as follows, see Fig. 14.

Experimental results show that disabling any component will lead to significant performance degradation.

In the case of single-component disabling, the performance was the worst when the emission phase was disabled (179.61), highlighting the critical role of solution evaluation. The repair phase also plays an important role, as removing it drops the performance to 167.65.

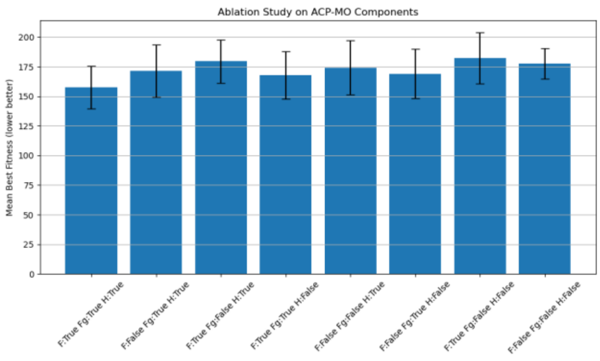

Fig. 14. Code output image 14

Furthermore, the configuration with multiple components disabled shows a significant drop in performance, further verifying that each module contributes necessary functionality. In summary, this ablation study validates that the formation, emission, and repair modules are essential for the robustness and efficiency of the ACP-MO algorithm.

3.6 Real-World Case Analysis: Pressure Vessel Design Problem

In order to verify the applicability of the ACP-MO algorithm in real-world engineering optimization problems, this paper selects the classic pressure vessel design problem as a test case. This problem is widely used in engineering design optimization research. The goal is to minimize the manufacturing cost of a cylindrical pressure vessel while satisfying a series of structural and volume constraints.

The evaluation process of ACP-MO in this problem also covers the three stages of "forming", "firing" and "healing", corresponding to design exploration, performance evaluation and detail optimization.

In the experiment, the population size is set to 30, the maximum number of iterations is 100, and the initial temperature is 1.0. During the execution process, the algorithm successfully found a set of optimal design parameters that meet all constraints, and the corresponding manufacturing cost is the lowest. The optimization results are as follows:

The optimal design variable values are:

x_1 = 12.597, x_2 = 82.720, x_3 = 55.634, x_4 = 152.085

The corresponding minimum manufacturing cost is: 773126.042.

This result verifies that ACP-MO has good performance and adaptability in dealing with complex, multi-constrained engineering optimization problems, and further supports its practical application potential in real-world tasks such as engineering design.

The complete source code for the proposed ACP-MO algorithm is available at the following link [34].

4. DISCUSSION

4.1 Algorithm Advantages

Compared to other traditional metaheuristic algorithms, ACPMO exhibits significant advantages in global search ability and convergence speed. For example, compared with genetic algorithms (GA) and particle swarm optimization algorithms (PSO), the sintering stage of ACPMO can effectively select high-quality solutions while maintaining solution diversity, thereby converging to the global optimal solution faster. Genetic algorithms (GA) and particle swarm optimization algorithms (PSO) usually have limited convergence speed and accuracy, and are prone to falling into local optimal solutions, especially in complex nonlinear multi-peak problems. ACPMO solves these problems by integrating operations such as molding, sintering, and repair, with stronger global search capabilities and faster convergence speed.

Additionally, ACPMO exhibits relatively low computational complexity and demonstrates high efficiency in solving practical engineering problems. Its high efficiency is not only reflected in its search ability, but also in its stability and reliability when dealing with large-scale problems.

4.2 Future Work

Future research should explore more complex parameter optimization methods, explore the cooling strategy of ACPMO in more depth, and, in addition to parameter optimization, ACPMO can also be integrated with other intelligent algorithms, and the application of ACPMO in multi-objective optimization problems should be further explored.

5. CONCLUSION

This paper proposes a ceramic process-based heuristic optimization algorithm (ACPMO), which shows good performance in various classic optimization problems. By simulating the molding, sintering and repairing steps inherent in the ceramic manufacturing process, ACPMO can effectively perform global optimization and overcome the common defects of traditional methods, such as easily falling into local optimal solutions in complex problem scenarios.

Experimental results show that ACPMO has strong global search capabilities and fast convergence speed, and can provide high-quality solutions for complex optimization problems. Compared with existing mainstream optimization algorithms, ACPMO not only effectively alleviates the problem of local optimal solutions but also converges to the global optimum faster and more accurately. It is worth noting that ACPMO shows very competitive performance in Rastrigin and Levi functions, outperforming many traditional algorithms (genetic algorithm (GA), evolutionary algorithm (DE), functional analysis (FA), ant colony algorithm (ACO) and autocorrelation algorithm (SA)), while the performance difference with compressed sensing (CS) is not significant. This shows that ACPMO performs well in dealing with multi-peak complex optimization scenarios, with high accuracy and robustness, but for some other functions, ACPMO does not outperform some other advanced algorithms, which shows that although ACPMO is effective, it may have difficulties in dealing with certain functions, especially when the gradient is not clear near the global optimal value, there are multiple local optimal values, and it is relatively flat.

CONFLICT OF INTEREST

The author declares no conflict of interest.

List of Abbreviations Used in the Paper

Acronym	Full Name
ACPMO	A novel metaheuristic optimization algorithm based on an advanced ceramic processing metaphor for optimization
GA	Genetic Algorithm
PSO	Particle Swarm Optimization
DE	Differential Evolution
CS	Cuckoo Search
FA	Firefly Algorithm
ACO	Ant Colony Optimization
SA	Simulated Annealing
WOA	Whale Optimization Algorithm
COA	Coati Optimization Algorithm
AOA	Aquila Optimizer
PDO	Perceptual Difference Optimizer

REFERENCES

[1] E. Trojovská, M. Dehghani, P. Trojovský, Zebra Optimization Algorithm: A New Bio-Inspired Optimization Algorithm for Solving Optimization Algorithm. *IEEE Access*, 10, 2022: 49445–49473. https://doi.org/10.1109/ACCESS.2022.3172789

[2] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, A.H. Gandomi, The Arithmetic Optimization Algorithm. *Computer Methods in*

Applied Mechanics and Engineering, 376, 2021: 113609.
https://doi.org/10.1016/j.cma.2020.113609

[3] D.E. Finkel, DIRECT Optimization Algorithm User Guide. *Center for Research in Scientific Computation, North Carolina State University*, 2(1), 2003: 1-14.

[4] P.C. Fourie, A.A. Groenwold, The Particle Swarm Optimization Algorithm in Size and Shape Optimization. *Structural and Multidisciplinary Optimization*, 23(4), 2002: 259–267.
https://doi.org/10.1007/s00158-002-0188-0

[5] A.-b. Meng, Y.-c. Chen, H. Yin, S.-z. Chen, Crisscross Optimization Algorithm and Its Application. *Knowledge-Based Systems*, 67, 2014: 218–229.
https://doi.org/10.1016/j.knosys.2014.05.004

[6] M. Khishe, M.R. Mosavi, Chimp Optimization Algorithm. *Expert Systems with Applications*, 149, 2020: 113338.
https://doi.org/10.1016/j.eswa.2020.113338

[7] D. Wang, D. Tan, L. Liu, Particle Swarm Optimization Algorithm: An Overview. Soft Computing, 22(2), 2018: 387–408.
https://doi.org/10.1007/s00500-016-2474-6

[8] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium Optimizer: A Novel Optimization Algorithm. *Knowledge-Based Systems*, 191, 2020: 105190.
https://doi.org/10.1016/j.knosys.2019.105190

[9] T. Rahkar Farshi, Battle Royale Optimization Algorithm. *Neural Computing and Applications*, 33, 2021: 1139–1157.
https://doi.org/10.1007/s00521-020-05004-4

[10] Q. Bai, Analysis of Particle Swarm Optimization Algorithm. *Computer and Information Science*, 3(1), 2010: 180.
https://doi.org/10.5539/cis.v3n1p180

[11] Y. Zhang, S. Wang, G. Ji, A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering*, 2015(1), 2015: 931256.
https://doi.org/10.1155/2015/931256

[12] M. Ghaemi, M.R. Feizi-Derakhshi, Forest Optimization Algorithm. *Expert Systems with Applications*, 41(15), 2014: 6676–6687.
https://doi.org/10.1016/j.eswa.2014.05.009

[13] A.G. Gad, Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review. *Archives of Computational Methods in Engineering*, 29, 2022: 2531–2561.
https://doi.org/10.1007/s11831-021-09694-4

[14] F.H. Zhou, Z.Z. Liao, A Particle Swarm Optimization Algorithm. *Applied Mechanics and Materials*, 303-306, 2013: 1369–1372.
https://doi.org/10.4028/www.scientific.net/AMM.303-306.1369

[15] R. Rao, Jaya: A Simple and New Optimization Algorithm for Solving Constrained and Unconstrained Optimization Problems. *International Journal of Industrial Engineering Computations*, 7(1), 2016: 19–34.
https://doi.org/10.5267/j.ijiec.2015.8.004

[16] T. Liao, T. Stützle, M.A.M. de Oca, M. Dorigo, A Unified Ant Colony Optimization Algorithm for Continuous Optimization. *European Journal of Operational Research*, 234(3), 2014: 597–609.
https://doi.org/10.1016/j.ejor.2013.10.024

[17] L. Abualigah, D. Yousri, M. Abd Elaziz, A.A. Ewees, M.A. Al-Qaness, A.H. Gandomi, Aquila Optimizer: A Novel Meta-Heuristic Optimization Algorithm. *Computers & Industrial Engineering*, 157, 2021: 107250.
https://doi.org/10.1016/j.cie.2021.107250

[18] A.R. Mehrabian, C. Lucas, A Novel Numerical Optimization Algorithm Inspired from Weed Colonization. *Ecological Informatics*, 1(4), 2006: 355–366.
https://doi.org/10.1016/j.ecoinf.2006.07.003

[19] M. Khishe, M. Nezhadshahbodaghi, M.R. Mosavi, D. Martín, A Weighted Chimp Optimization Algorithm. *IEEE Access*, 9, 2021: 158508–158539.
https://doi.org/10.1109/ACCESS.2021.3130933

[20] H. Jia, H. Rao, C. Wen, S. Mirjalili, Crayfish Optimization Algorithm. *Artificial Intelligence Review*, 56(Suppl 2), 2023: 1919–1979.
https://doi.org/10.1007/s10462-023-10567-4

[21] F.A. Hashim, K. Hussain, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, Archimedes Optimization Algorithm: A New Metaheuristic Algorithm for Solving Optimization Problems. *Applied Intelligence*, 51, 2021: 1531–1551.
https://doi.org/10.1007/s10489-020-01893-z

[22] Y. Shi, An Optimization Algorithm Based on Brainstorming Process. In: Emerging Research on Swarm Intelligence and Algorithm Optimization. *IGI Global*, 2015: 1–35.

https://doi.org/10.4018/978-1-4666-6328-2.ch001

[23] J.E. Onwunalu, L.J. Durlofsky, Application of a Particle Swarm Optimization Algorithm for Determining Optimum Well Location and Type. *Computational Geosciences,* 14, 2010: 183–198.
https://doi.org/10.1007/s10596-009-9142-1

[24] M.H. Amiri, N. Mehrabi Hashjin, M. Montazeri, S. Mirjalili, N. Khodadadi, Hippopotamus Optimization Algorithm: A Novel Nature-Inspired Optimization Algorithm. Scientific Reports, 14(1), 2024: 5032.
https://doi.org/10.1038/s41598-024-54910-3

[25] S. Bandyopadhyay, S. Saha, U. Maulik, K. Deb, A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA. *IEEE Transactions on Evolutionary Computation*, 12(3), 2008: 269–283.
https://doi.org/10.1109/TEVC.2007.900837

[26] B. Abdollahzadeh, N. Khodadadi, S. Barshandeh, P. Trojovský, F.S. Gharehchopogh, E.-S.M. El-kenawy, L. Abualigah, S. Mirjalili, Puma Optimizer (PO): A Novel Metaheuristic Optimization Algorithm and Its Application in Machine Learning. *Cluster Computing*, 27(4), 2024: 5235–5283.
https://doi.org/10.1007/s10586-023-04221-5

[27] M. Dehghani, Z. Montazeri, E. Trojovská, P. Trojovský, Coati Optimization Algorithm: A New Bio-Inspired Metaheuristic Algorithm for Solving Optimization Problems. *Knowledge-Based Systems,* 259, 2023: 110011.
https://doi.org/10.1016/j.knosys.2022.110011

[28] A.E. Ezugwu, J.O. Agushaka, L. Abualigah, S. Mirjalili, A.H. Gandomi, Prairie Dog Optimization Algorithm. *Neural Computing and Applications*, 34, 2022: 20017–20065.
https://doi.org/10.1007/s00521-022-07530-9

[29] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris Hawks Optimization: Algorithm and Applications. *Future Generation Computer Systems*, 97, 2019: 849–872.
https://doi.org/10.1016/j.future.2019.02.02

[30] S. He, Q.H. Wu, J.R. Saunders, Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior. *IEEE Transactions on Evolutionary Computation*, 13(5), 2009: 973–990.
https://doi.org/10.1109/TEVC.2009.2011992

[31] B. Abdollahzadeh, F.S. Gharehchopogh, A Multi-Objective Optimization Algorithm for Feature Selection Problems. *Engineering with Computers*, 38(Suppl 3), 2022: 1845–1863.
https://doi.org/10.1007/s00366-021-01369-9

[32] T.S. Ayyarao, N.S.S. Ramakrishna, R.M. Elavarasan, N. Polumahanthi, M. Rambabu, G. Saini, B. Khan, B. Alatas, War Strategy Optimization Algorithm: A New Effective Metaheuristic Algorithm for Global Optimization. *IEEE Access*, 10, 2022: 25073–25105.
https://doi.org/10.1109/ACCESS.2022.3153493

[33] J.S. Pan, L.G. Zhang, R.B. Wang, V. Snášel, S.C. Chu, Gannet Optimization Algorithm: A New Metaheuristic Algorithm for Solving Engineering Optimization Problems. *Mathematics and Computers in Simulation*, 202, 2022: 343–373.
https://doi.org/10.1016/j.matcom.2022.06.007

[34] https://github.com/JJJJGOOD/ACP-MO” (Access: 23 June 2025)