Article

Not peer-reviewed version

High-speed 3D printing temperature control with optimized PID parameters based on improved ALO algorithm

Rui Zhou * , Junchi Jiang , Chunzhi Du , Shuaiyuan Lu

Posted Date: 23 January 2024

doi: 10.20944/preprints202401.1664.v1

Keywords: (FDM; improved ant-lion optimization algorithm (OCALO); composite chaotic mapping; Elite Opposition-Based learning; cosine factors.)

Article

High-Speed 3D Printing Temperature Control with Optimized PID Parameters Based on Improved ALO Algorithm

Rui Zhou *, **Junchi Jiang**, **Chunzhi Du** and **Shuaiyuan Lu**

College of Transportation Science and Engineering, Civil Aviation University of China, Tianjin 300300, China
* Correspondence: reaterbutter@163.com

Abstract: To address the large hysteresis and susceptibility to instability in high-speed deposition fusion moulding (FDM) 3D printing's temperature control, this paper proposes an improved ant-lion optimization algorithm (OCALO) to optimize the PID parameters.The improvement aims to solve the problem that the traditional ant-lion optimization algorithm (ALO) reduces the population diversity and tends to fall into the local optimal solution when it is close to the global optimal solution.The generation of the initial solution is enhanced by the introduction of a new Tent-Logistic-Cotangent composite chaotic mapping. This integration into Elite Opposition-Based Learning aids in optimizing population diversity. Concurrently, the incorporation of cosine factors of a specified parameter into the elitist formula of the traditional ALO algorithm is aimed at reducing the tendency of ants to randomly gravitate towards ant-lions with lower fitness values. This adjustment helps in balancing the exploration-exploitation trade-off in the algorithmic process. Compared with two existing classical algorithms and three improved ALO algorithms, the algorithm improves the convergence speed, global search ability and the ability to jump out of the local optimal solution. The improved algorithm was combined with PID control to design an OCALO-PID temperature controller and simulated on a high-speed 3D printing temperature model identified by modelling. The results show that the method improves the transient and steady-state performance of reactor temperature control with good control accuracy and robustness. Finally, the proposed algorithm is applied to a physical experimental platform to verify the feasibility of the algorithm.

Keywords: FDM; improved ant-lion optimization algorithm (OCALO); composite chaotic mapping; Elite Opposition-Based learning; cosine factors

1. Introduction

Fused deposition modeling (FDM) technology, a widely used and promising three-dimensional (3D) printing technology [1], is rapidly growing. High-speed FDM 3D printing is favored for its fast printing speed, networked control, parameter visualization, among other features. During the use of high-speed FDM 3D printing, though, it is found that it takes a relatively lengthy period of time for users to wait for the nozzle to heat up, and that real-time monitoring of the print nozzle temperature jitter is becoming increasingly serious in the 3D printing process.

The quality of 3D printed part is significantly affected by the speed of printing and nozzle temperature [2]. Although with optimization algorithms such as fuzzy control [3], model predictive control [4] and robust control [5] have emerged in the field of 3D printing temperature control with the development of technology and advanced control strategies, The temperature control of FDM 3D printing in practical applica-tions is still dominated by the proportional-integral-derivative (PID) control meth-od,which is simple in principle as well as in structure, and is thereby widely used for a wide range of problems: process control, magnetic and optic memories, automo-tive,etc [6]. Traditional PID control methods have certain limitations in the face of complex nonlinear systems due to the nonlinearity and hysteresis of the actual temperature during the heating and printing process of the FDM 3D printing nozzle.

The ant-lion algorithm (ALO),as a new type of intelligent algorithm with good robustness and strong optimization-seeking ability, can effectively solve the stability problems occurring in the temperature control of high-speed FDM 3D printing,and has been adopted in various professions [7–9]. Nevertheless, the ALO algorithm needs to be improved because it still has the problem of relying too much on the initial solution, the elitist efficiency is not high, and it is easy to fall into the local optimization. Nowa-days, there are scholars who have achieved superior outcomes by improving the ALO algorithm for diverse tasks. Li [10] and others developed a Gaussianmutation based ALO (GALO) algorithm and applied it to the 0-1 backpack problem. In order to accurately and efficiently diagnose the health state of rolling bearings, Wang [11] proposed a rolling bearing fault diagnosis model based on Improved Antli-on Optimization (IALO) algorithm to optimize BP neural network. Jie Ji [12] et al. in-vestigated the equipment capacity allocation of a chemical company's cogeneration system for cooling, heating and power generation based on the improvement of ant-lion optimization algorithm by increasing the weight coefficients(WALO).

Considering the high requirements of high-speed FDM 3D printing on temperature control accuracy and stability, a 3D printing nozzle temperature control algorithm based on an improved ant-lion optimisation algorithm to optimize the PID parameters is presented in response to the problems in the above literature. Unlike other improved algorithms, the algorithm not only improves it from the perspectives of population initialisation and elitist formulae, which improves its search accuracy, convergence speed and stability, but also improves the existing chaotic mapping and proposes a Tent-Logistic-Cotangen composite mapping method, which effectively overcomes the characteristics of ALO that is prone to fall into local optimum and does not increase the time complexity of the algorithm, which has certain novelty. Finally, the method is applied to the MatLab high-speed 3D printing temperature simulation experiment and the semi-physical platform based on the Klipper firmware, which verifies the effectiveness and feasibility of the method.

2. Proposed Optimization

2.1. Traditional ALO

Ant Lion Optimizer (Ant Lion Optimizer) [13] is a natural heuristic algorithm proposed by Australian scholar Seyedali inspired by the hunting mechanism of ant li-ons in nature The core idea of the algorithm is to simulate the hunting mechanism of ant lions hunting ants in order to achieve the global optimization. The ant lion utilizes its traps dug into the sand in the shape of a funnel and hides at the bottom of the trap to wait for the hunting objects to approach the trap, once the ants engaging in random wandering behaviors fall into the trap, the ant lion would imme-diately capture and feed on them. After predation is complete, the funnel-shaped trap will be repaired and the next predation will take place.Interventionary studies involv-ing animals or humans, and other studies that require ethical approval, must list the authority that provided approval and the corresponding ethical approval code. As the movement of ants is stochastic, random walking is used to simulate the mobility of ants and the mathematical formulas describing the coordinates of their activities are as follows:

$$\mathrm{X}_{\mathrm{i,j}}^{\mathrm{t}} = [0, \text{ cumsum } \left(2r\left(t_1\right) - 1\right), \ldots, \text{ cumsum } \left(2r\left(t_n\right) - 1\right)] \tag{1}$$

$$r(t) = \begin{cases} 1 & \text{if rand } > 0.5 \\ 0 & \text{if rand } \leqslant 0.5 \end{cases} \tag{2}$$

where $\mathrm{X}_{\mathrm{i,j}}^{\mathrm{t}}$is the position of each individual ant, t represents the current number of iterations, $cumsum()$ denotes the cumulative sum of the randomized wandering steps,n is the maximum number of wandering times,$t_n$is the current number of iterations, and $r(t)$ is a random function, $rand$ is the random number, and $rand \in [0, 1]$. Ant lions update their position to the prevailing location of the

doi:10.20944/preprints202401.1664.v1

ant being hunted to enhance their opportunities to capture fresh pre-existing prey. The ant-lion is constantly updating its position through Equation (3).

$$\begin{aligned} &\text{Antlion}\,_i^t = \text{ Ant}\,_i^t \\ &\text{if } f\left(\text{ Ant}\,_i^t\right) > f\left(\text{ Antlion}\,_i^t\right) \end{aligned} \tag{3}$$

where f $\left(\text{ Ant}\,_i^t\right)$ and $f\left(\text{ Antlion}_i^t\right)$ are the fitness values of the ith-dimensional ants and ant-lion at the tth iteration, and Ant_i^t and $\text{Antlion}_i^t$are the positions of the ith-dimensional ants and ant lion at the tth iteration. The Elite Strategy ensures that the optimal solution is obtained at any stage of the optimization process, and the best adapted ant lion in each iteration is reserved as the Elite Ant Lion, and each ant would select an Elite Ant Lion depending on the Roulette Strategy and the Elite Strategy, around which it would wandering randomly. Ants keep updating their position through equation (4).

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{4}$$

where$R_A^t$is the position where the ants randomly walk around the antlion selected by roulette, and$R_E^t$is the position where the ants randomly walk around the elite antlion.

Ant Lion Optimizer (ALO) is a robust intelligent algorithm inspired by the hunting mechanism of ant lions in nature. Despite its strengths, it has drawbacks such as a lack of population diversity and a tendency to converge to local optima, necessitating improvements for more effective applications.

2.2. Classical chaotic mappings initialize populations

The traditional ALO algorithm employed random initialisation of the population, which tends to produce an inhomogeneous distribution of the initialised population and is not conducive to covering further areas in the search space. By taking a chaotic initialisation method to initialise the population, the coverage as well as the diversity can be increased. At present, the most commonly used methods of chaotic initialisation are logistic mapping, sine mapping,tent mapping and so on.

$$\text{Logistic map :} \quad x_{i+1} = 4rx_i\left(1 - x_i\right) \tag{5}$$

$$\text{Sine map:} \quad x_{i+1} = r\sin\left(\pi x_i\right) \tag{6}$$

$$\text{Tent map :} \quad x_{i+1} = \begin{cases} 2rx_i & \text{if } x_i < 0.5 \\ 2r\left(1 - x_i\right) & \text{if } x_i \geq 0.5 \end{cases} \tag{7}$$

Combining a diversity of low-dimensional chaos to form a new composite chaotic system. This type of chaotic system can effectively overcome the shortcomings of low-dimensional chaos, and is less complex and can be realised more conveniently than high-dimensional chaos.Tent-Logistic-Cosine chaos mapping utilises the existing Logistic, Sine and Tent maps as seed maps [14] as in Equation(8).

$$x_{i+1} = \begin{cases} \cos\left(\pi\left(2rx_i + 4(1-r)x_i\left(1 - x_i\right) - 0.5\right)\right) & \text{for } x_i < 0.5; \\ \cos\left(\pi\left(2r\left(1 - x_i\right) + 4(1-r)x_i\left(1 - x_i\right) - 0.5\right)\right) & \text{for } x_i \geq 0.5, \end{cases} \tag{8}$$

The distribution (left) and histogram (right) of the 1000 generations of Tent-Logistic-Cosine chaos mapping were generated on MATLAB as shown in Figure 2

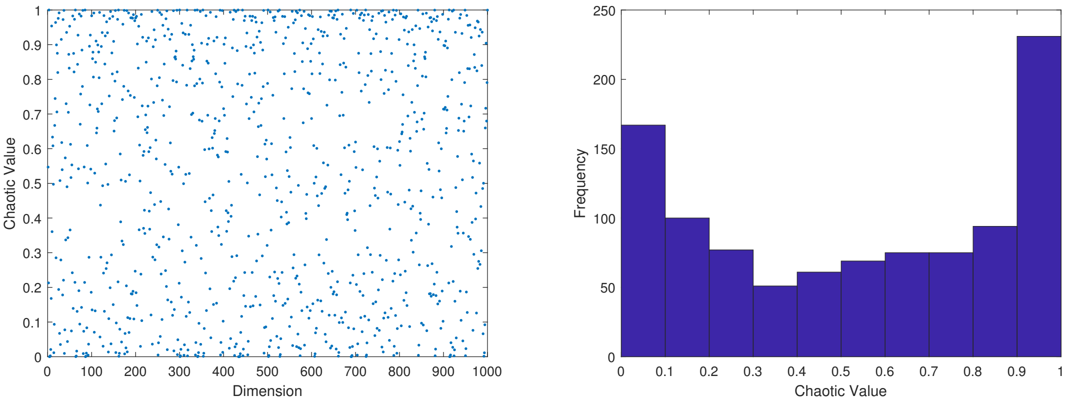


Figure 1. (**a**)Chaotic sequence distribution diagram. (**b**) Histogram of distribution of chaotic sequence

The above figure shows that the sequence generated by the iteration of Tent-Logistic-Cosine chaotic mapping is not distributed uniformly in the phase space, meanwhile its parameter space in the chaotic state is narrower, and the chaotic effect is less effective.

2.3. Improved ALO

To address the above problems of traditional ALO, this paper improves the algorithm by introducing improved composite chaotic mapping and Elite Opposition-Based learning to initiate the population and increase the diversity of the population.

2.3.1. Tent-Logistic-Cotangent composite chaotic mapping

To avoid the existence of small circles and unstable points in the chaotic mapping, the periodicity, ergodicity and regularity of the chaotic variables are maintained.In this paper, we present a new chaotic mapping, named Tent-Logistic-Cotangent composite mapping, with the mathematical expression as in equation (9).

$$x_{i+1} = \begin{cases} \bmod\left(\left(\pi\left(rcot(4\pi x_i) + 4(1-r)x_i\left(1-x_i\right) - 0.2\right)\right), 1\right) & \text{for } x_i < 0.5; \\ \bmod\left(\left(\pi\left((1-r)cot(4\pi x_i) + 4(1-r)x_i\left(1-x_i\right) - 0.15\right)\right), 1\right) & \text{for } x_i \geq 0.5, \end{cases} \tag{9}$$

In contrast to equation (8),the distribution (left) and histogram (right) of the 1000 generations of Tent-Logistic-Cotangent chaos mapping were also generated on MATLAB as shown in Figure 2.

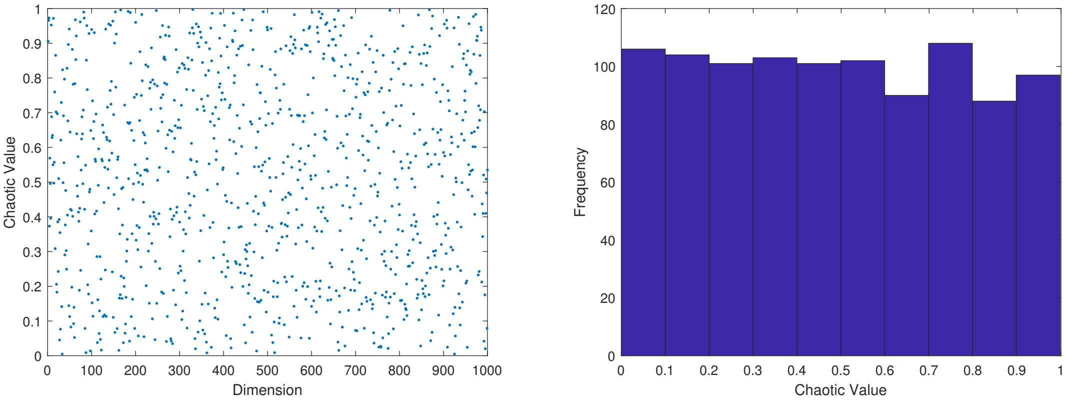


Figure 2. (**a**)Tent-Logistic-Cotangent chaotic sequence distribution diagram. (**b**) Histogram of distribution of chaotic sequence

Comparing Figures 1 and 2, the proposed Tent-Logistic-Cotangent composite chaotic mapping iteration produces sequences that are more uniformly distributed in the phase space with excellent ergodicity.

2.3.2. Elite Opposition-Based learning

Opposition-Based Learning (OBL) is a machine intelligence strategy proposed by Tizhoosh [15] in 2005, and is used effectively to improve the quality of the solution by applying it to initialize the population. Elite Opposition-Based learning (EOBL) can improve on the basis of Opposition-Based Learning [16] , which extracts the elite individuals in the population in order to calculate the reverse individuals, and then select the candidate solutions with superior fitness values from the candidates.

Due to the great chance and blindness of the solutions generated by chaotic mapping, the generated solutions are subjected to Elite Opposition-Based learning to save convergence time and improve the quality of the solutions.It has been shown that real-world engineering problems have been solved by Elite Opposition-Based learning [17] .

Thus the elite backward learning based update initialization of the formulation as in equation (10).

$$X_{i,j}^{t+2} = k * (lb + ub) - X_{i,j}^{t+1} \tag{10}$$

where $k \in [0,1]$. lb is the lower bound of the solution space. ub is the upper bound of the solution space.

2.3.3. Introduction of Cosine Factor Strategy

Traditional ALO algorithms later stage ant individuals perform random wandering towards ill-adapted ant lions, which can lead to weak iterative results and local optimum stagnation problems.Therefore, the cosine factor is introduced into the elitist update formula for adjusting the search direction to be closer to the direction of the optimization goal, and the algorithm is more exploratory, better avoids falling into the local optimum, and approaches the global optimum solution faster during the search process.Inspired by the cultural-particle swarm optimization algorithm (C-PSO) [18] , the algorithm uses adaptive inertia weights to balance global and local search capabilities.

Based on this, this paper designs a cosine factor for a given parameter can effectively reduce the situation of falling into the local optimum, and improve the search ability of local development and global,as in equations (11)-(12).

$$\theta(k) = \omega_{\min} * (1 - \cos h) + \omega_{\max} * \cos h \tag{11}$$

$$\varphi(k) = r * \cos h + \omega_{\max} * \cos h \tag{12}$$

$$AntP_i^t = \theta^2(k) R_A^t + \varphi^2(k) R_E^t \tag{13}$$

where $h = \pi t / 2t_{\max}$. $t_{\max}$represents the maximum number of iterations. $\omega_{\max} = 0.5$,$\omega_{\min} = 0.1$.

3. Performance Analysis on Benchmark Functions

3.1. Selection ofTest Functions

In order to verify the feasibility of the improved antlion algorithm, 23 different types of test functions [19] were selected for simulation experiments to examine the full range of the improved antlion algorithm's optimization-seeking ability through different types of functions. Among them, F1 to F7 are high-dimensionalunimodal test functions; F8 to F13 are high-dimensional multimodal test functions; F14 to F23 are fixed-dimensional multimodal test functions.As shown in Table 1. In the experiment, it was chosen to set all the population numbers to 50, and the maximum number of iterations were all 300 times.

Table 1. Test function.

Type	Function Name	Dimensionality	Search Space	Optimum Value
High-dimensional unimodal	Sphere(F1)	30	$[-100, 100]$	0
	Schwefel 2.22(F2)	30	$[-10, 10]$	0
	Schwefel 1.2(F3)	30	$[-100, 100]$	0
	Schwefel 2.21(F4)	30	$[-100, 100]$	0
	Generalized Rosenbrock(F5)	30	$[-30, 30]$	0
	Step Function(F6)	30	$[-100, 100]$	0
	Quartic(F7)	30	$[-1.28, 1.28]$	0
High-dimensional multimodal	Schwefel2.26(F8)	30	$[-500, 500]$	-418.9829*30
	Rastrigin(F9)	30	$[-5.12, 5.12]$	0
	Ackley(F10)	30	$[-32, 32]$	0
	Griewank(F11)	30	$[-600, 600]$	0
	Generalized Penalized Function 1(F12)	30	$[-50, 50]$	0
	Generalized Penalized Function 2(F13)	30	$[-50, 50]$	0
Fixed-dimensional multimodal	Shekel's Foxholes(F14)	2	$[-65.536, 65.536]$	1
	Kowalik(F15)	4	$[-5, 5]$	0.0003
	Six-Hump Camel-Back(F16)	2	$[-5, 5]$	-1.0316
	Branin(F17)	2	$[-5, 5]$	0.398
	Goldstein-Price(F18)	2	$[-2, 2]$	3
	Hartman's Family n = 3(F19)	3	$[0, 1]$	-3.86
	Hartman's Family n = 6(F20)	6	$[0, 1]$	-3.32
	Shekel's Family m = 5(F21)	4	$[0, 10]$	-10.1532
	Shekel's Family m = 7(F22)	4	$[0, 10]$	-10.4028
	Shekel's Family m = 10(F23)	4	$[0, 10]$	-10.5363

3.2. Experimental Environment and Comparison Algorithm Selection

The experiments were conducted in AMD Ryzen 7 6800H CPU@3.20 GHz, 16.00 GB of memory, Windows 10 system and MATLAB R2020a.The original PSO algorithm, ALO algorithm, GALO algorithm, IALO algorithm, and WALO algorithm were compared for benchmarking functions. Among them, the selection of each comparison algorithm is based on the following foundations: ALO algorithm is the original algorithm of OCALO; Particle Swarm algorithm has a simple structure and a short running time in the search process, which is used to focus on comparing the stability and real-time performance of OCALO algorithm in the search process; GALO algorithm, IALO algorithm, WALO algorithm, which are other improved algorithms with a wide range of searching scope and strong ability, and easy to find the optimal value of the test function. GALO algorithm, IALO algorithm, WALO algorithm and other improved algorithms have a wide search range and strong ability to find the optimal value of the test function.

3.3. Comparative Analysis ofPerformance Indicators

Take the average of the results of 50 iterations as in Table 2, Table 3 is the optimal value that can be achieved by different optimization algorithms in the number of iterations, Table 4 is the worst value that can be achieved by different optimization algorithms in the number of iterations, Table 5 is the sample variance of different optimization algorithms, and Table 6 is the number of iterations of the baseline test function that arrives at the result.

Table 2. Mean value of test function search results.

Function	OCALO	PSO	ALO	GALO	IALO	WALO
F1	**2.61E-12**	1.18E-04	2.22E-09	1.04E-09	3.49E-10	3.92E-10
F2	**2.12E-07**	1.07E-02	3.12E-04	5.46E-06	1.12E-05	1.22E-05
F3	**1.94E-11**	3.17E-02	2.89E-06	8.29E-09	1.31E-09	2.51E-09
F4	**3.27E-07**	2.25E-02	9.67E-05a	3.21E-05	8.78E-06	3.83E-06
F5	**1.07E-03**	9.17	6.08	3.46E-03	8.42E-03	1.25E-03
F6	1.62E-05	4.83E-05	**8.19E-07**	1.27E-05	2.88E-04	4.73E-03
F7	**5.78E-06**	1.09E-02	4.38E-03	1.33E-04	1.23E-04	2.43E-04
F8	**-6978.02**	-3112.81	-3521.99	-5478.15	-6205.49	-5645.51
F9	**-2.83E-12**	10.88	18.24	5.32	9.37E-02	1.43E-10
F10	**1.29E-06**	1.55E-02	3.74E-01	1.50E-04	7.17E-06	2.33E-06
F11	**5.06E-12**	1.97E-01	3.45E-01	2.18E-02	6.64E-09	2.95E-09
F12	**1.67E-06**	8.58E-04	25.94	2.67E-05	8.61E-04	4.20E-05
F13	**6.29E-06**	3.49E-03	5.79E-03	3.45E-05	7.14E-05	1.32E-05
F14	**0.998**	0.9821	1.3282	0.9978	1.3979	1.0613
F15	**3.10E-04**	7.82E-04	8.89E-04	3.38E-04	3.20E-04	3.43E-04
F16	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	-1.0018
F17	**0.39789**	0.3983	0.3982	0.3964	0.3977	0.4121
F18	**3**	**3**	**3**	**3**	3.00012	3.35123
F19	**-3.8625**	-3.8628	-3.8628	-3.8643	-3.8662	-3.8508
F20	**-3.3216**	-3.2198	-3.2187	-3.1892	-3.1718	-3.0714
F21	**-10.016**	-9.751	-8.362	-10.015	-8.681	-3.237
F22	**-10.3985**	-9.7201	-6.7462	-5.1766	-8.5914	-4.3017
F23	**-9.5311**	-8.8172	-3.6639	-7.3814	-7.9204	-4.3954

Table 3. Optimal value of test function search results.

Function	OCALO	PSO	ALO	GALO	IALO	WALO
F1	**1.02E-12**	2.81E-05	1.11E-09	4.49E-10	1.85E-10	8.50E-11
F2	**5.09E-08**	5.81E-05	1.37E-05	3.79E-06	1.07E-05	7.77E-06
F3	**7.14E-12**	2.16E-02	1.57E-07	2.82E-09	6.91E-10	1.58E-09
F4	**5.19E-08**	2.02E-02	4.13E-05	1.99E-05	6.42E-06	2.04E-06
F5	**9.95E-05**	6.10	2.64E-03	1.70E-03	1.91E-03	5.80E-04
F6	1.46E-05	1.94E-05	**5.89E-10**	6.19E-06	7.20E-05	1.16E-04
F7	**2.26E-06**	8.91E-03	3.67E-03	6.99E-05	6.77E-06	9.07E-06
F8	**-12564.63**	-6979.09	-5614.97	-8274.44	-12341.18	-8803.05
F9	**6.68E-13**	7.31	7.96	7.99	3.91E-09	1.24E-10
F10	**4.15E-07**	7.25E-03	3.12E-05	1.31E-05	4.06E-05	1.37E-06
F11	**1.36E-12**	1.16E-01	1.38E-01	1.40E-08	1.27E-10	2.07E-10
F12	**1.01E-07**	1.29E-06	4.51E-01	6.99E-06	1.51E-06	1.85E-06
F13	**5.67E-06**	2.08E-05	1.74E-5	1.66E-05	2.82E-05	5.96E-06
F14	**0.998**	**0.998**	**0.998**	**0.998**	**0.998**	0.99811
F15	**3.02E-04**	7.68E-03	1.22E-03	3.08E-04	3.09E-04	3.18E-04
F16	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	-1.0247
F17	**0.39789**	**0.39789**	**0.39789**	**0.39789**	0.39791	0.40085
F18	**3**	**3**	**3**	**3**	**3**	3.0436
F19	**-3.8615**	-3.8628	-3.8628	-3.8628	-3.8628	-3.8481
F20	**-3.3205**	-3.3220	-3.3220	-3.3220	-3.3213	-3.0163
F21	**-10.1532**	**-10.1532**	**-10.1532**	**-10.1532**	-10.11091	-4.1752
F22	**-10.4029**	**-10.4029**	**-10.4029**	-10.3991	-10.3893	-5.3595
F23	**-10.5363**	-10.5364	-5.1756	-10.5361	-10.4548	-7.5633

Table 4. Worst value of test function search results.

Function	OCALO	PSO	ALO	GALO	IALO	WALO
F1	**3.05E-12**	2.25E-04	3.16E-09	2.78E-09	4.17E-10	3.92E-10
F2	**2.38E-07**	1.79E-02	7.43E-01	6.81E-06	1.17E-05	1.37E-05
F3	**1.09E-10**	6.81E-02	1.02E-05	9.92E-09	4.38E-09	3.08E-09
F4	**8.23E-07**	3.28E-02	5.52E-04	3.21E-05	8.43E-06	4.34E-06
F5	**1.12E-03**	14.90	9.61	5.11E-03	1.64E-02	3.05E-03
F6	2.98E-05	1.51E-04	**3.01E-06**	1.56E-05	3.05E-04	7.72E-03
F7	**9.21E-06**	1.77E-02	5.11E-02	1.77E-03	3.94E-04	2.16E-04
F8	**-4182.9987**	-1925.8472	-2044.2853	-3970.2836	-2213.5056	-4053.0627
F9	**6.10E-12**	12.08	30.84	13.98	3.75	1.62E-10
F10	**1.29E-06**	3.10E-2	1.12	1.01E-2	7.17E-05	3.13E-06
F11	**9.72E-12**	1.97E-01	3.45E-01	2.19E-01	2.84E-08	2.95E-09
F12	**8.51E-06**	2.14E-02	97.07	5.37E-05	1.75E-04	1.09E-04
F13	**9.69E-06**	2.60E-01	1.12E-02	7.48E-04	7.07E-04	4.60E-05
F14	**0.998**	1.992	1.992	3.9683	0.9979	1.2267
F15	**3.10E-04**	9.90E-04	1.00E-03	3.38E-04	2.19E-03	3.53E-04
F16	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	-1.0247
F17	**0.39789**	0.3997	0.3987	0.3904	0.39721	0.4254
F18	**3**	**3**	**3**	**3**	3.0004	3.6114
F19	**-3.8627**	-3.8628	-3.8628	-3.8628	-3.7486	-3.8014
F20	**-3.3219**	-2.6829	-3.2031	-3.1462	-3.0634	-2.8063
F21	**-5.0551**	-2.6829	-5.055	-2.6829	-2.6295	-1.9851
F22	**-10.3451**	-5.1288	-2.7519	-1.8376	-3.7234	-2.3182
F23	**-5.5386**	-5.1756	-2.4217	-1.6766	-2.4273	-2.0791

Table 5. Sample variance of test function search results.

Function	OCALO	PSO	ALO	GALO	IALO	WALO
F1	**1.07E-24**	1.35E-08	1.08E-18	1.36E-18	1.46E-20	5.31E-20
F2	**1.07E-14**	3.71E-05	2.28E-08	2.61E-12	3.97E-13	9.87E-12
F3	**3.03E-21**	6.07E-04	2.56E-11	1.54E-11	3.48E-18	5.63E-19
F4	**1.53E-13**	3.99E-05	8.14E-08	3.75E-11	5.34E-12	1.33E-12
F5	**2.28E-06**	23.02	29.34	2.96E-06	5.38E-06	8.94E-06
F6	6.61E-11	4.37E-09	**2.42E-12**	2.56E-11	1.41E-08	1.91E-05
F7	**1.66E-11**	2.83E-04	5.79E-04	7.50E-07	2.68E-08	1.25E-08
F8	23410890.62	**6510698.73**	3954183.82	9148403.26	31725496.17	13017423.61
F9	**7.37E-24**	5.81	144.86	30.98	3.51	3.66E-22
F10	**2.61E-13**	1.74E-04	1.28E-02	3.28E-07	1.06E-09	6.73E-13
F11	**1.82E-23**	1.53E-04	1.38E-01	1.12E-02	2.62E-16	1.90E-18
F12	**4.38E-12**	1.29E-06	2247.12	5.85E-10	1.14E-08	2.53E-09
F13	**4.36E-12**	2.16E-02	4.17E-05	1.72E-07	1.32E-07	4.04E-10
F14	**0**	3.29E-01	2.96E-01	2.29	3.29E-01	1.13E-02
F15	**1.88E-11**	1.50E-08	2.89E-08	7.59E-10	1.17E-07	3.16E-10
F16	**0**	**0**	**0**	**0**	**0**	1.87E-04
F17	**0**	8.19E-07	1.64E-07	1.40E-07	1.75E-07	1.13E-04
F18	**0**	**0**	**0**	**0** 3.58E-08	5.77E-02	
F19	3.67E-07	**0**	**0**	**0**	2.43E-03	4.25E-04
F20	**4.87E-07**	4.71E-03	4.72E-03	5.60E-03	1.69E-02	1.93E-03
F21	8.66	9.88	9.96	40.99	18.67	**1.42**
F22	**9.45E-05**	11.91	30.85	22.10	11.04	2.32
F23	**11.49**	14.37	2.66	38.25	20.74	8.99

Table 6. Number of iterations to test the optimization results of the function.

Function	OCALO	PSO	ALO	GALO	IALO	WALO
F1	**292**	295	299	300	293	294
F2	**279**	300	294	300	298	294
F3	**287**	300	300	300	300	300
F4	**296**	300	300	298	297	300
F5	118	300	152	300	**83**	87
F6	212	300	300	300	209	**117**
F7	132	300	141	177	**87**	165
F8	23	300	**15**	211	79	19
F9	300	**31**	66	71	300	273
F10	**290**	300	300	297	292	296
F11	300	230	254	**228**	300	294
F12	**135**	300	300	300	282	142
F13	181	300	243	300	300	**121**
F14	29	16	23	33	**11**	30
F15	152	109	232	113	211	**65**
F16	197	123	111	111	57	**6**
F17	52	26	32	50	92	**17**
F18	128	84	32	54	206	**27**
F19	202	197	56	152	229	**8**
F20	242	262	225	274	206	**24**
F21	123	171	156	121	75	**22**
F22	242	222	150	250	275	**26**
F23	200	253	226	229	254	**4**

3.4. Comparison ofConvergence Curves ofFitness Values

In order to reflect the dynamic convergence characteristics of OCALO and to further visually compare the convergence of the algorithms and the ability to jump out of the local optimum, the convergence curves of the 23 test functions are given for a total of 23 test functions under the five optimization algorithms, where the horizontal coordinates are the number of iterations and the vertical coordinates are the values of the fitness function. It is shown in Figures 3–5.

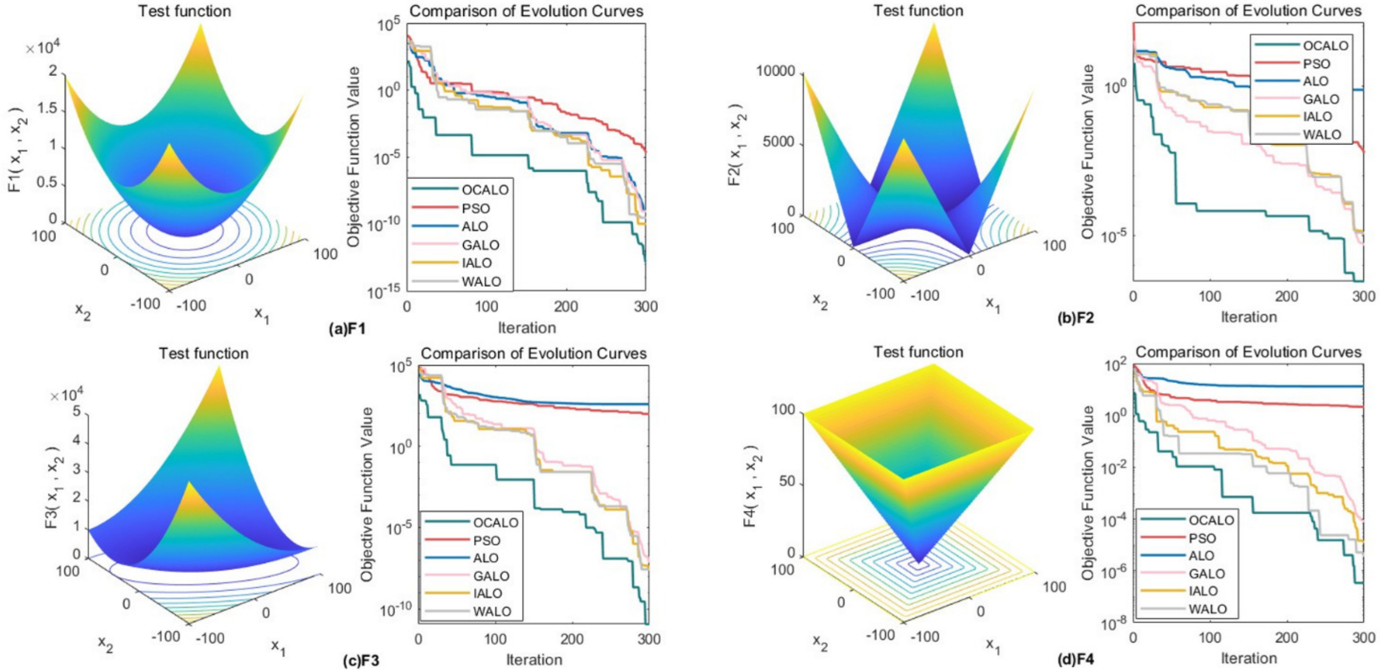
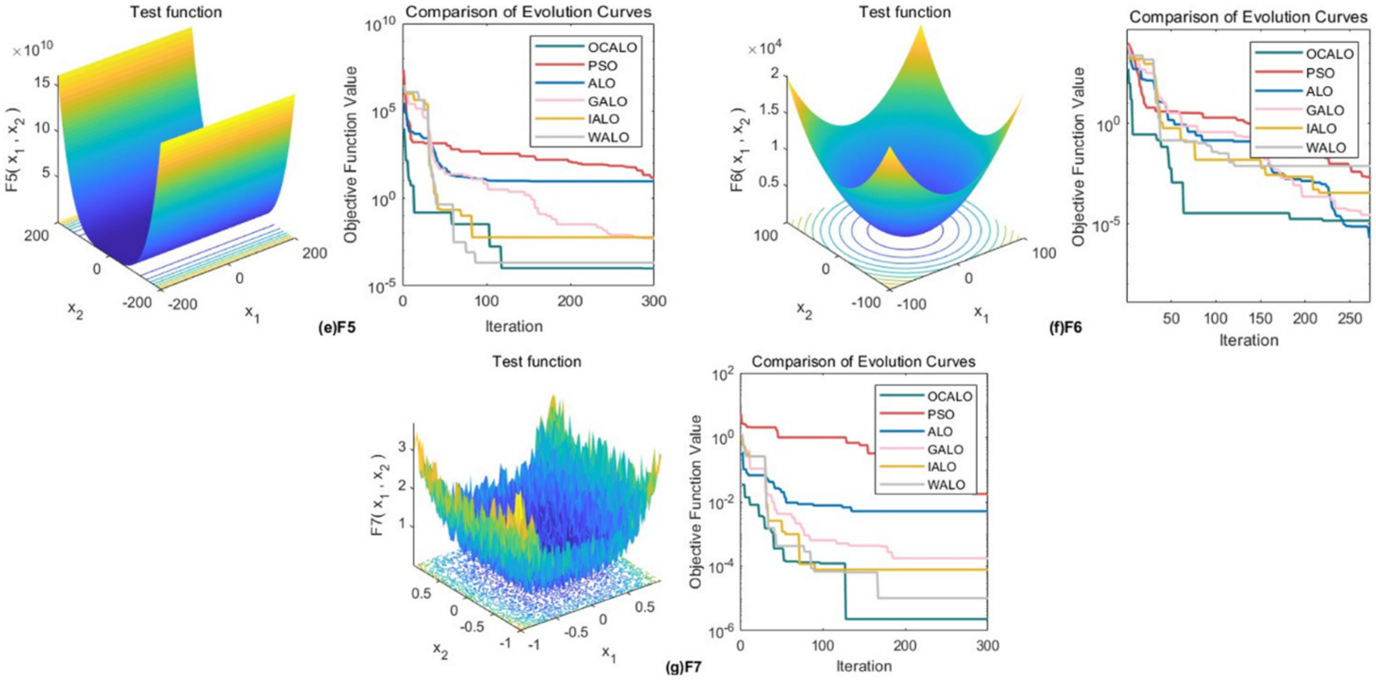


Figure 3. High-dimensional F1-F4 function convergence curve.

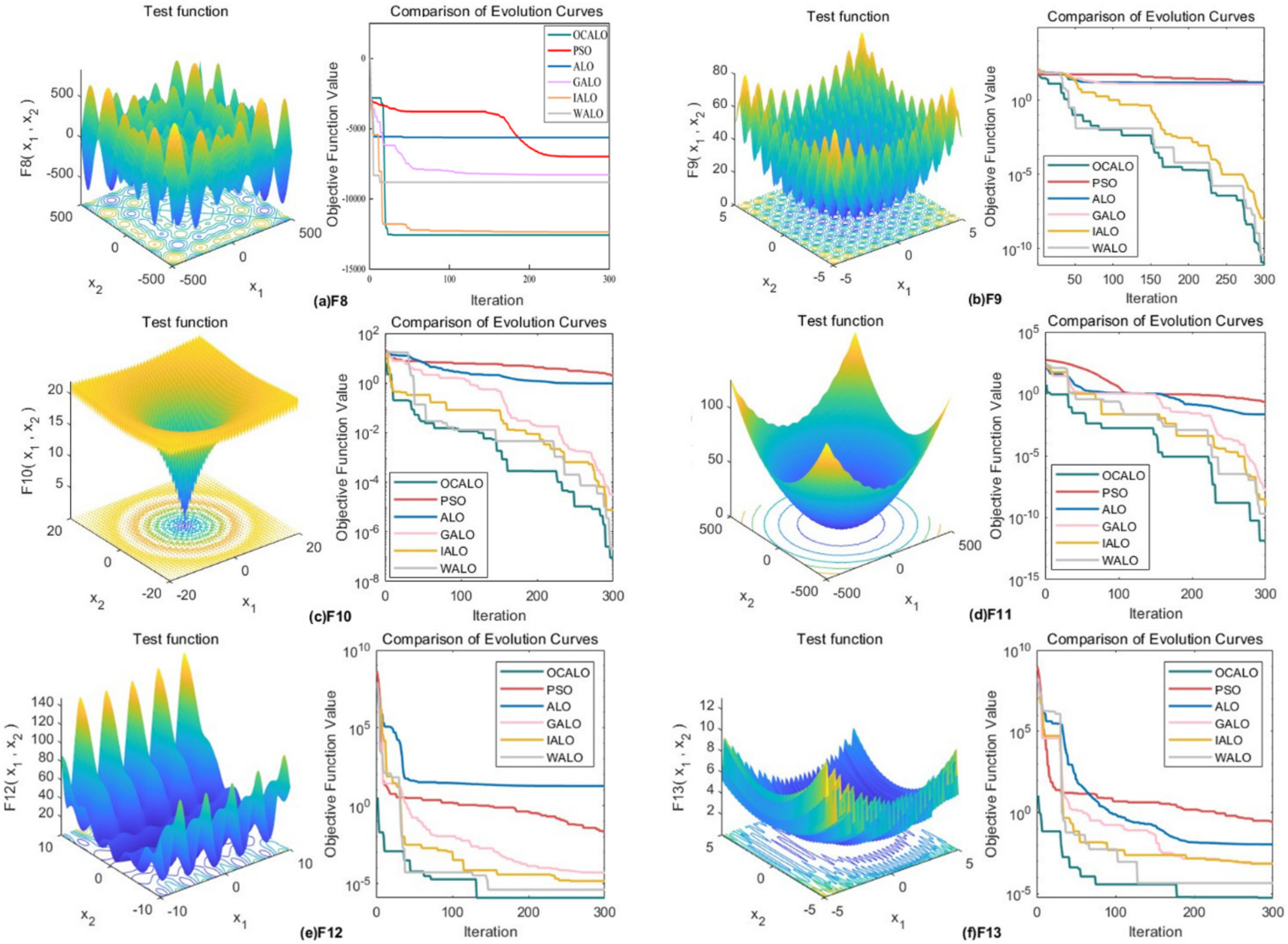


(e)F5

(f)F6

(g)F7

Figure 4. High-dimensional single-peak F5-F7 function convergence curve.



(a)F8 (b)F9 (c)F10 (d)F11 (e)F12 (f)F13

Figure 5. High-dimensional multi-peak functions convergence curve.

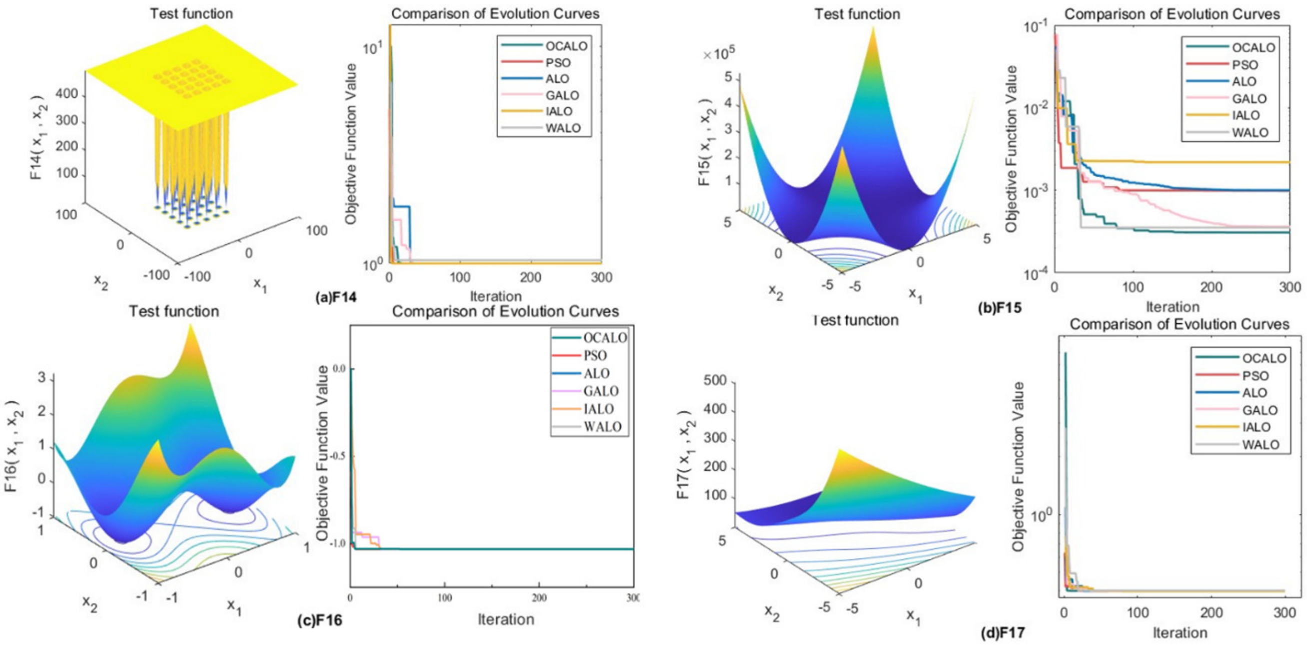


Figure 6. *Cont*.

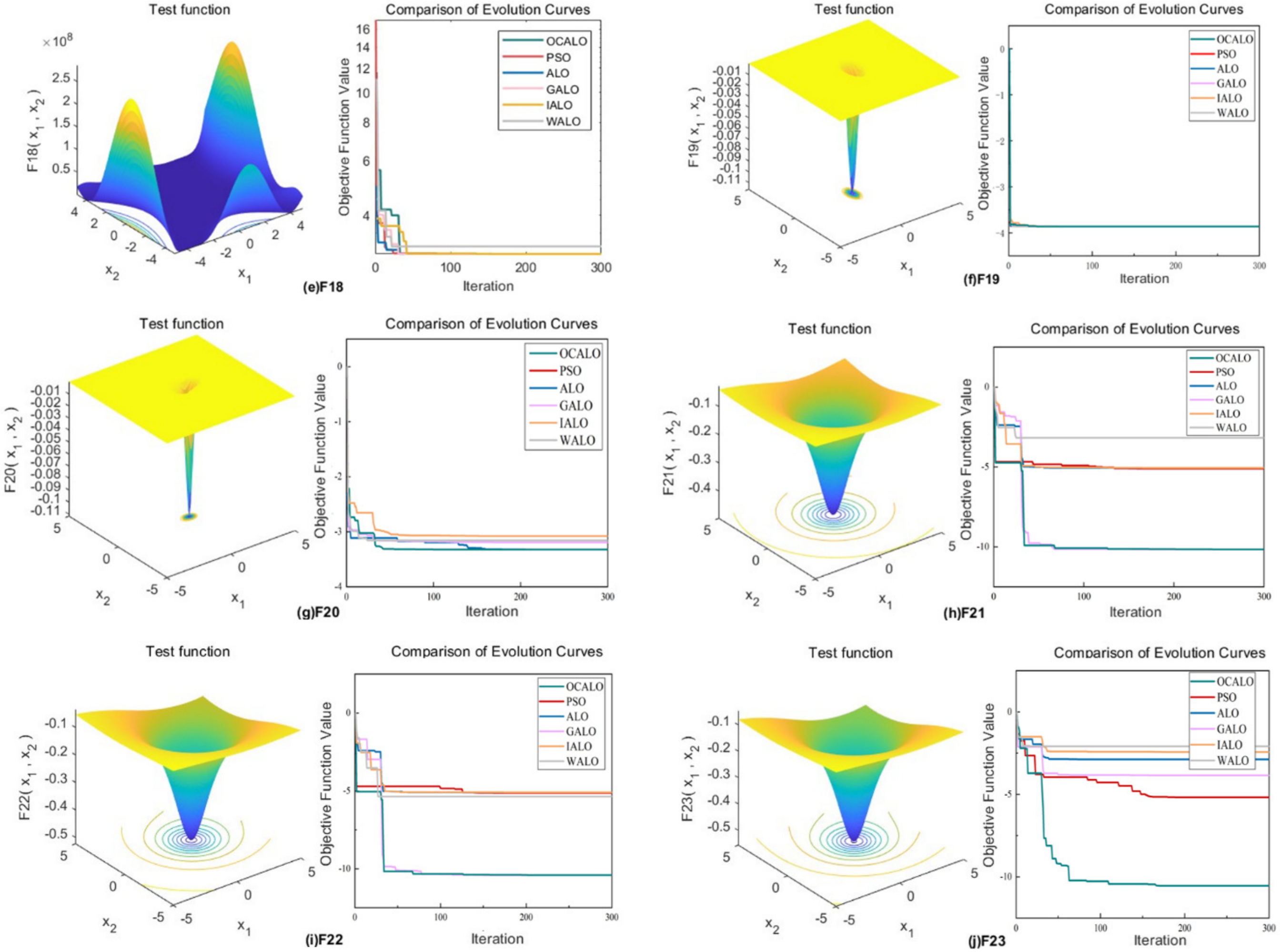


Figure 6. Fixed -dimensional multimodal function convergence curve.

In summary, it can be seen that OCALO has significantly improved the search ability for different types of benchmark functions, whether it is a high-dimensional single-peak function, high-dimensional multi-peak function, fixed-dimensional multi-peak function. At the same time, its fast convergence speed and short operation time can meet the demand for real-time algorithms and effectively avoid falling into local optimum while ensuring the search speed, thus proving the feasibility and superiority of OCALO.

4. Performance Analysis of High-speed 3D printing model for Temperature Control

4.1. High-speed 3D printing Model Identificatio

The 3D printing nozzle heating time versus temperature obtained after running for 500 seconds is shown in Table 7 for some of the data.The real-time temperature profile of the high-speed FDM 3D printing nozzle will have a small oscillating effect.

Table 7. 3D printing nozzle real-time temperature.

Time(s)	0	1	...	300	301	...	499	500
Temp(°C)	30.4	30.9	...	209.8	209.9	...	210	209.9

Considering that due to the modelling process of the 3D printing nozzle and its complexity, the transfer function of the first order inertial delay system is difficult to capture the complex dynamic behaviour of the nozzle, in order to conveniently analyse and control the dynamic response of the whole system, the form of the transfer function of the system is designed as a second order model as shown in Equation (14).

$$G(s) = \frac{k\omega^2 e^{-\theta s}}{s^2 + 2\xi\omega s + \omega^2} \tag{14}$$

where k is system gain. ω is the Frequency. ξ is the Damping Factor.θ is the Delay.

The values of the parameters were obtained by fitting the output curves as shown in Figure 7.

Based on the fitted response curve in Table 8, the process transfer function can be derived as Equation (15).Since the temperature system of 3D printing is a typical time delay system, in this paper, the time delay is set to 20.

$$G(s) = \frac{1.014}{s^2 + 0.05s + 0.001} e^{-20s} \tag{15}$$

Table 8. Model Parameters.

Parameters	**Notation**	**Value**
Gain	k	859.663
Frequency	ω	0.034
Damping Factor	ξ	0.737
Delay	θ	0

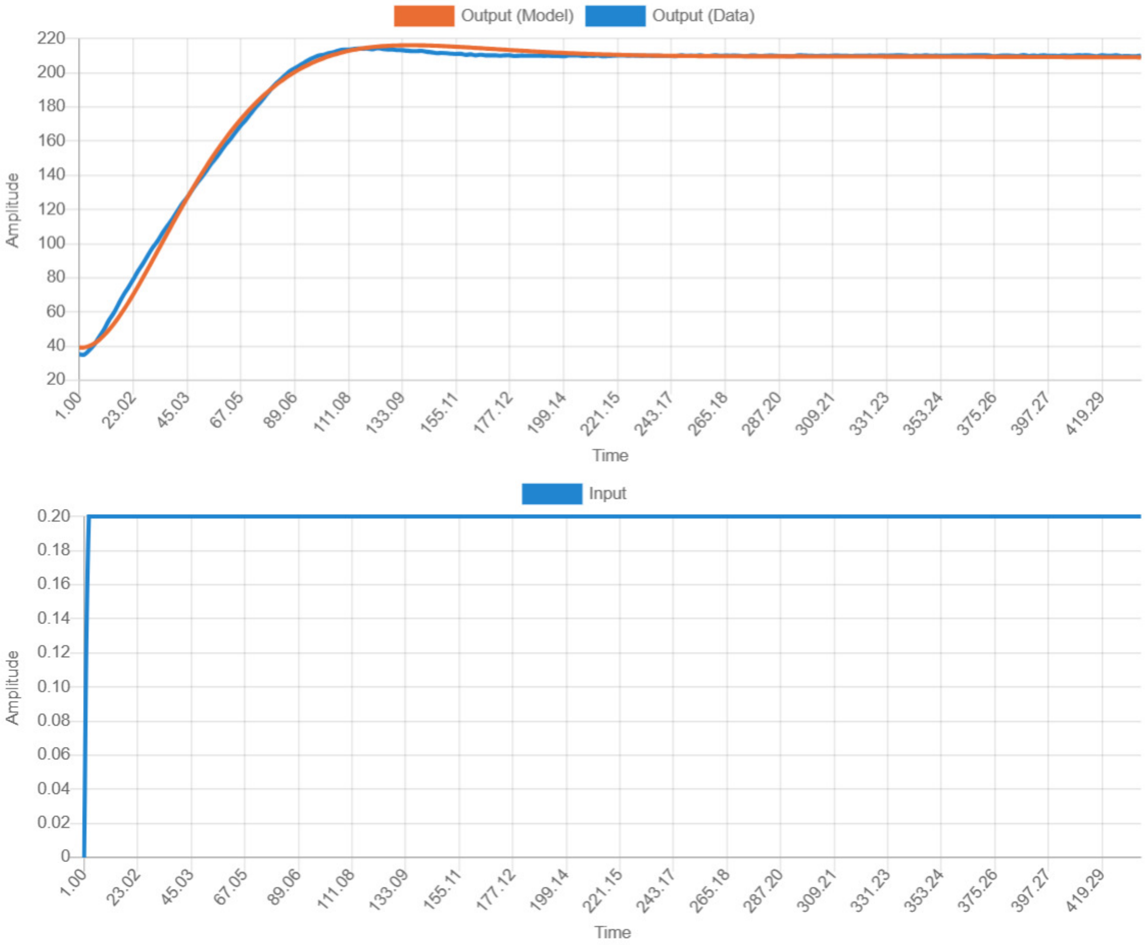


Figure 7. Fit Model.

4.2. Controller Design

The quality of the parameters in the PID controller will greatly affect the quality of the controller. In this paper, we design an optimal PID controller based on the improved Ant-Lion optimization algorithm, with the aim of finding a set of optimal parameters in the solution space K_p, K_d,K_i, so that the system meets the control requirements and performs well. In designing the OCALO algorithm PID controller, the objective function should be set in accordance with the performance index of the control system and obtain satisfactory dynamic characteristics of the iterative process, and the evaluation function T is set as shown in equation (16).

$$T = \int_0^\infty \left(w_1|e(t)| + w_2 u^2(t) \right) \mathrm{d}t \tag{16}$$

where $e(t)$is the error of the output value with respect to the input value. $u(t)$ is the control value of the control margin. w_1 is the weight value, which takes the value [0, 1], and usually w_1 = 0.999. w_2 is the weight value, which takes the value [0, 1], and usually w_2 = 0.001. In addition, a restriction is needed to prevent the overshooting, i.e., when the overshooting occurs, an additional overshooting term is introduced in the objective function, and the overshooting term $w_3|e(t)|$is introduced in the objective function Q as shown in Equation (17).

$$Q = \int_0^\infty \left(w_1|e(t)| + w_2 u^2(t) + w_3|e(t)| \right) \mathrm{d}t \tag{17}$$

Where w_3 is the weight value,w_3»w_1,in general, w_3=100. The simulation model is built in simulink according to equation (15). The improved ant-lion optimization algorithm is used to design the PID controller, setting the parameter ranges $\mathrm{Kp}, \mathrm{Kd}, \mathrm{Ki} \in [0, 100000]$, and Equation (17) as the fitness function, and the goal of the optimization search is to find a set of PID values that minimise the Q error by correcting the three parameters through the OCALO algorithm. The block diagram of the PID controller design based on the OCALO algorithm is shown in Figure 9.

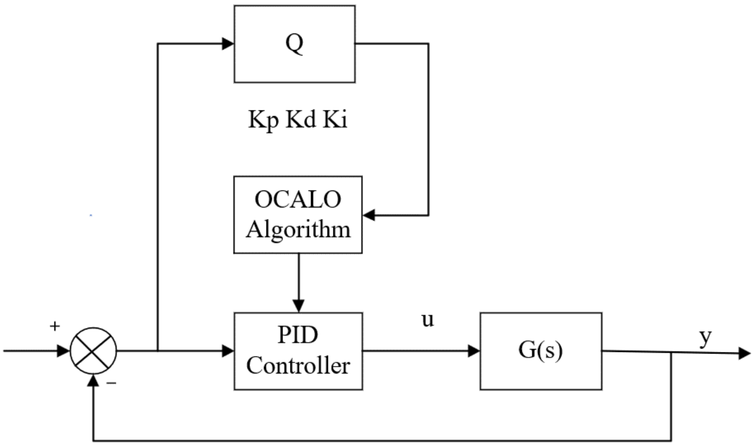


Figure 8. Block diagram of optimized PID controller design based on OCALO algorithm.

4.3. System Simulation and Results Analysis

In this paper, only OCALO algorithm is listed with PSO algorithm, ALO algorithm and IALO algorithm, because WALO algorithm and GALO algorithm optimize the PID of the model temperature system with a very slow convergence speed, so they do not draw their curves on the graph. From Figure 9, it is easy to see that the OCALO algorithm has a faster response, the least amount of overshooting and the fastest response when it optimizes the PID of the temperature system.

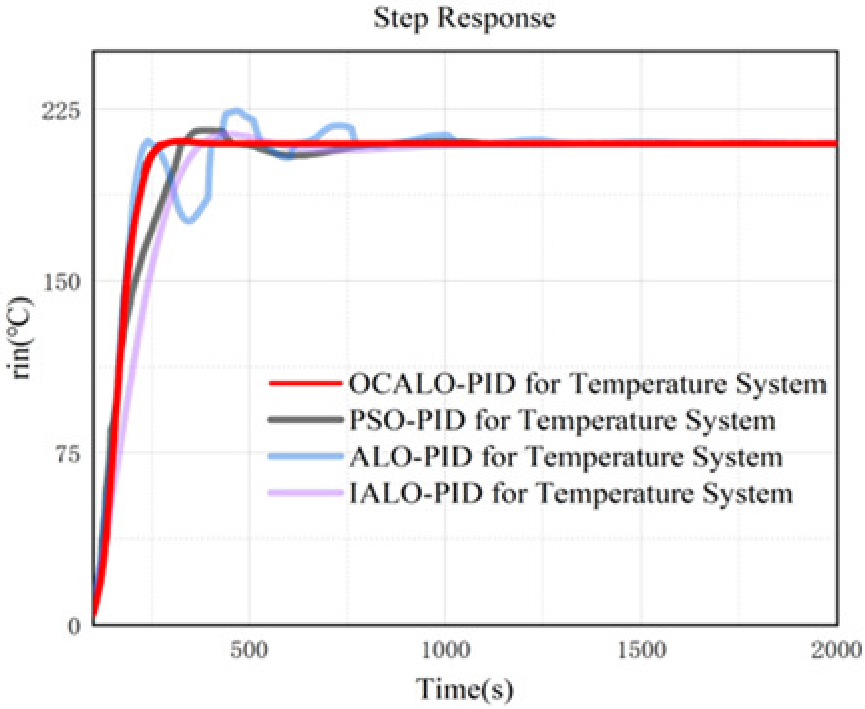


Figure 9. Partial algorithmic PID controller for temperature systems.

According to Table 8, through the peak amplitude, peak time, rise time, Settling time four aspects of the analysis, OCALO algorithm peak amplitude is the smallest indicates that the system in the process of PID rectification compared to other algorithms overshoot the smallest amount of the system is the most stable; the peak time as well as the shortest rise time indicates that the process of PID rectification compared to other algorithms with the shortest response time, the fastest response time; stabilisation time compared to other algorithms is the shortest, so that can be to a certain extent to save the high-speed printing time costs.

Table 9. Simulation results of PID controller.

Variable	OCALO	PSO	ALO	IALO
Peak amplitude	**1.59774**	5.7589	14.3121	4.0686
Peak time	**321.4505**	385.3550	469.4281	451.4830
Rise time	**272.0981**	328.0076	407.3485	379.7559
Settling time	**351.3188**	860.1506	619.9532	1040.6734

The results show that OCALO has stronger robust search ability and the convergence speed can meet the real-time requirements. After establishing the FDM 3D printing nozzle model and determining its parameters, the OCALO-PID controller is designed, and through experimental simulation, the control curves of the OCALO-PID controller and the classical PID controller are compared under different control requirements, and it is concluded that the overall control performance of the OCALO-PID controller is better.

4.4. Interference Test

In performing high-speed 3D printing, the 3D printing nozzle in the face of a sharp turn of the printing task, due to the rapid rotation will cause the 3D printing nozzle vibration, so that has been stabilised at the target temperature of the 3D printing nozzle temperature system sudden load, resulting in temperature instability, affecting the quality of the print, this paper adds a perturbation at 520s, the OCALO algorithm with the classical algorithm(left) and the other improvements of the ALO algorithm(right) comparison for example Figure 11.

Analyzing Figure 10, compare the time for various controllers to recover to within 2error bars after being disturbed by a 0.1 step signal at 520 s, as shown in Table 10. Comparing the immunity of each algorithm in simulation.Determine the immunity to interference in the simulation by both the maximum peak value and the time required for recovery.

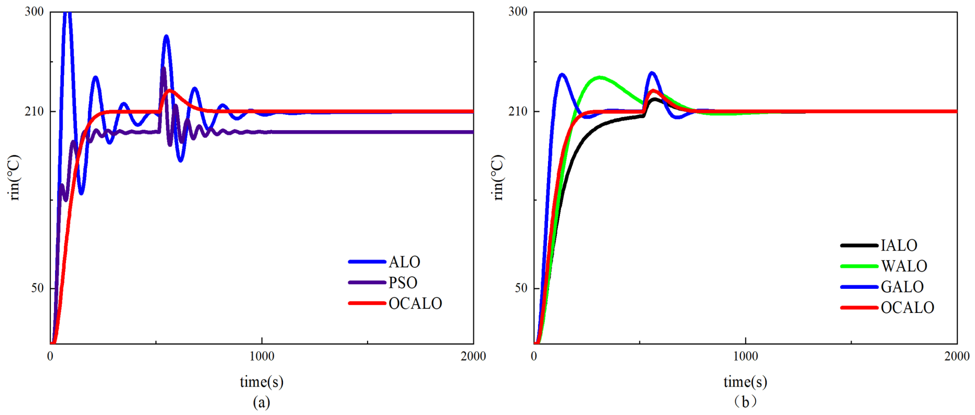


Figure 10. (**a**)Comparison of classical algorithms after sudden loading. (**b**) Comparison of Improved Algorithms after sudden loading.

Table 10. Recovery time.

Algorithm	Maximum peak value(°C)	The time required for recovery (s)
OCALO-PID	**219.3011**	**225.1**
PSO-PID	312.3896	-
ALO-PID	249.3161	-
GALO-PID	230.1576	247.5
IALO-PID	244.86196	244.2
WALO-PID	218.2119	242.9

After the perturbation at 520 s, OCALO-PID has the smallest maximum peak, PSO-PID and ALO-PID have already experienced oscillatory phenomena, and all other algorithms have larger peaks than OCALO-PID. the amplitudes of PSO-PID and ALO-PID are still outside of the2 % error bars at 1000 s, and the shortest time used by OCALO-PID is 225.1 s , 22.4 s faster than GALO-PID, 19.1 s faster than IALO-PID, and 17.8 s faster than WALO-PID, and the difference in the recovery time is smaller compared to SSA-PID, but it is also 10.8 s faster, verifying that the OCALO-PID controllers are more resistant to disturbances than the other controllers.The OCALO-PID controller shows good performance in simulation, but it needs to be validated by applying it to an actual high-speed 3D printing temperature system.

5. Performance Analysis on Physical Platform Validation

5.1. Experiment platform construction

In order to verify the control effect of OCALO-PID controller on the temperature system of high-speed FDM 3D printing, this paper builds an experimental platform as in Figure 11, and the upper computer is the web-side interface based on Klipper firmware. The real-time temperature monitoring module of the Klipper firmware provides feedback on the heating and vibration cancellation of the FDM 3D printing nozzle.

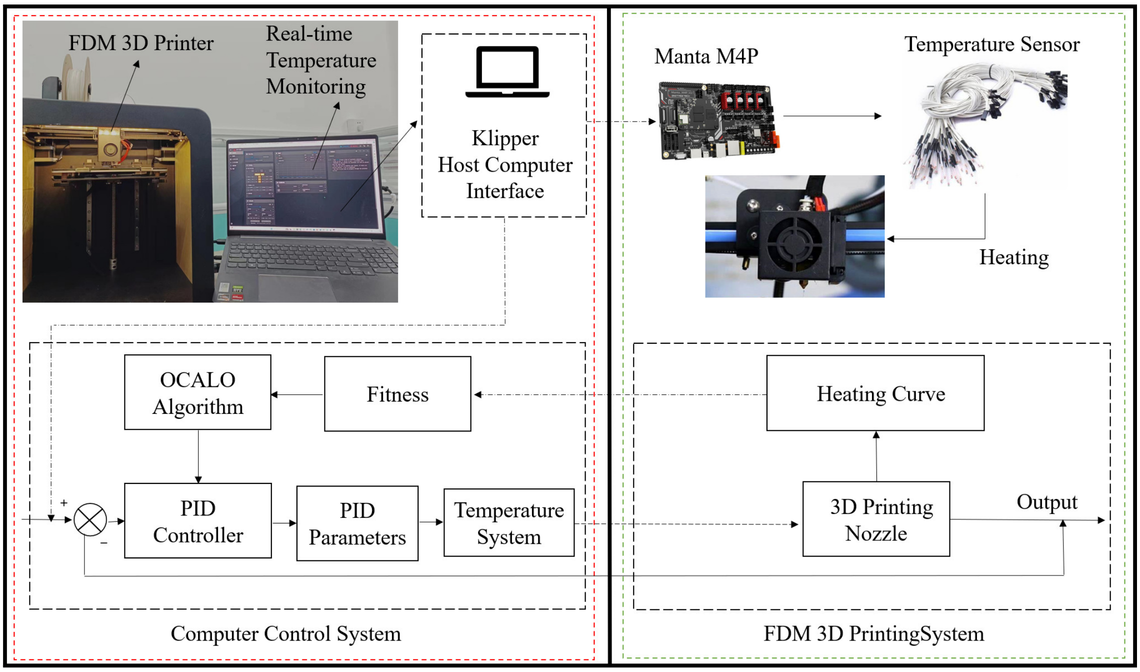


Figure 11. Setting up the experimental environment.

5.2. Response and stabilisation experiments

The desired temperature was set to 210 °C. The temperature of the 3D printing nozzle was controlled by PID parameters of the Klipper firmware. The temperature of the 3D printing was controlled by adding different algorithms. The temperature profile is shown in Figure 12.

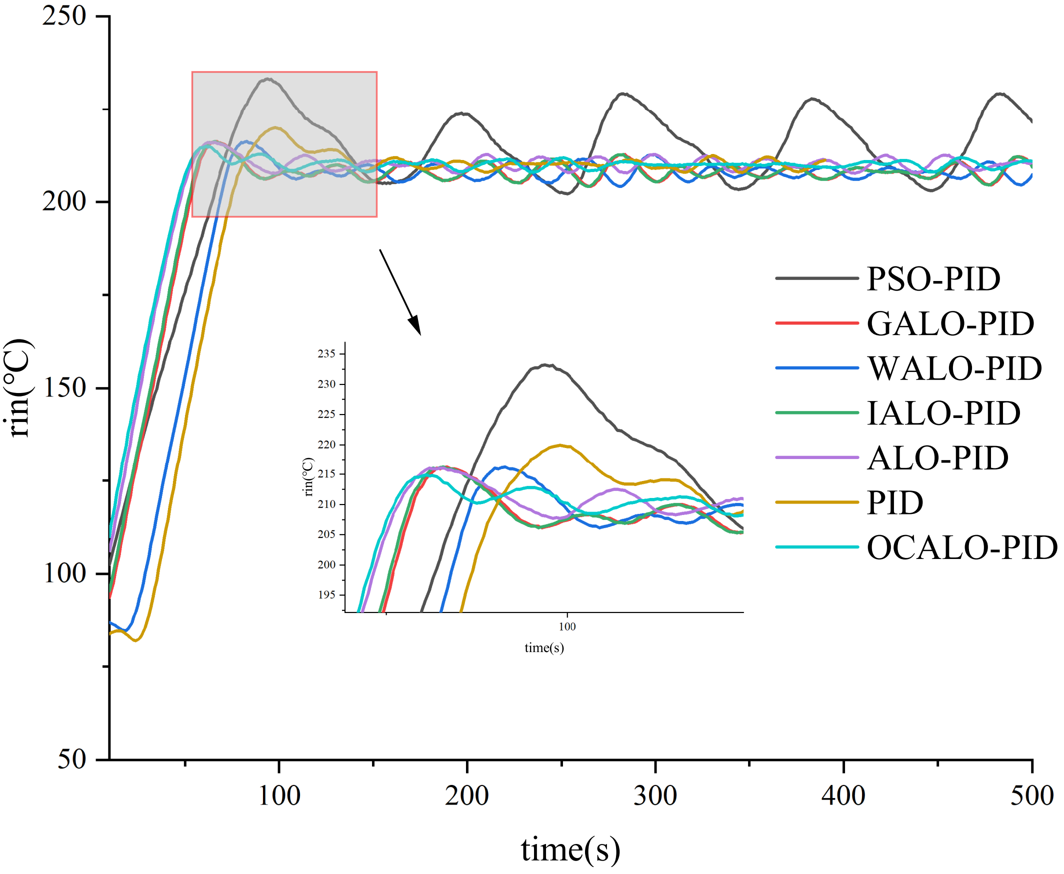


Figure 12. 3D printing temperature response curve.

As the actual 3D printing temperature will constantly jitter due to unavoidable reasons caused by the hardware, it is assumed in this paper that if there is a phase where the temperature jitter is less than 3°C, the temperature of the 3D printing printhead is considered stable.

Analysing Table 11, it can be seen that the OCALO algorithm not only achieves better responsiveness and robustness in the fitted simulation model, but also shows its good responsiveness and robustness in the actual high-speed 3D printing temperature system, and it can be seen that the OCALO algorithm achieves the fastest speed to reach the steady state and uses the least amount of actuality from Peak time and Stable time aspects.

Table 11. Performance index.

Algorithm	**Maximum Overshoot**(%)	**Peak time(s)**	**Stable time(s)**	K_p	K_i	K_d
PSO-PID	10.67	96	-	0	1.472	36.541
GALO-PID	2.95	68	276	27.251	0	3.637
WALO-PID	2.95	83	390	25.361	1.415	32.117
IALO-PID	2.95	68	239	21.679	1.441	107.504
ALO-PID	2.86	65	352	18.525	0.912	30.113
PID	4.71	99	353	14.699	0.572	94.068
OCALO-PID	**2.33**	**61**	**116**	20.837	1.037	104.708

5.3. Actual Interference Test

In real high-speed 3D printing application environments, different perturbations are often encountered,the cause of the perturbations may be due to human control requirements, or due to fast stepping movements, or may be indeterminate. Therefore, perturbation testing of algorithms is needed as a way to verify that the robustness of the algorithm meets the requirements. Perturbation of all algorithms.Add artificial perturbations in real environment 3D printing run 250 s. Figure 13 shows the perturbation test plot for all algorithms.

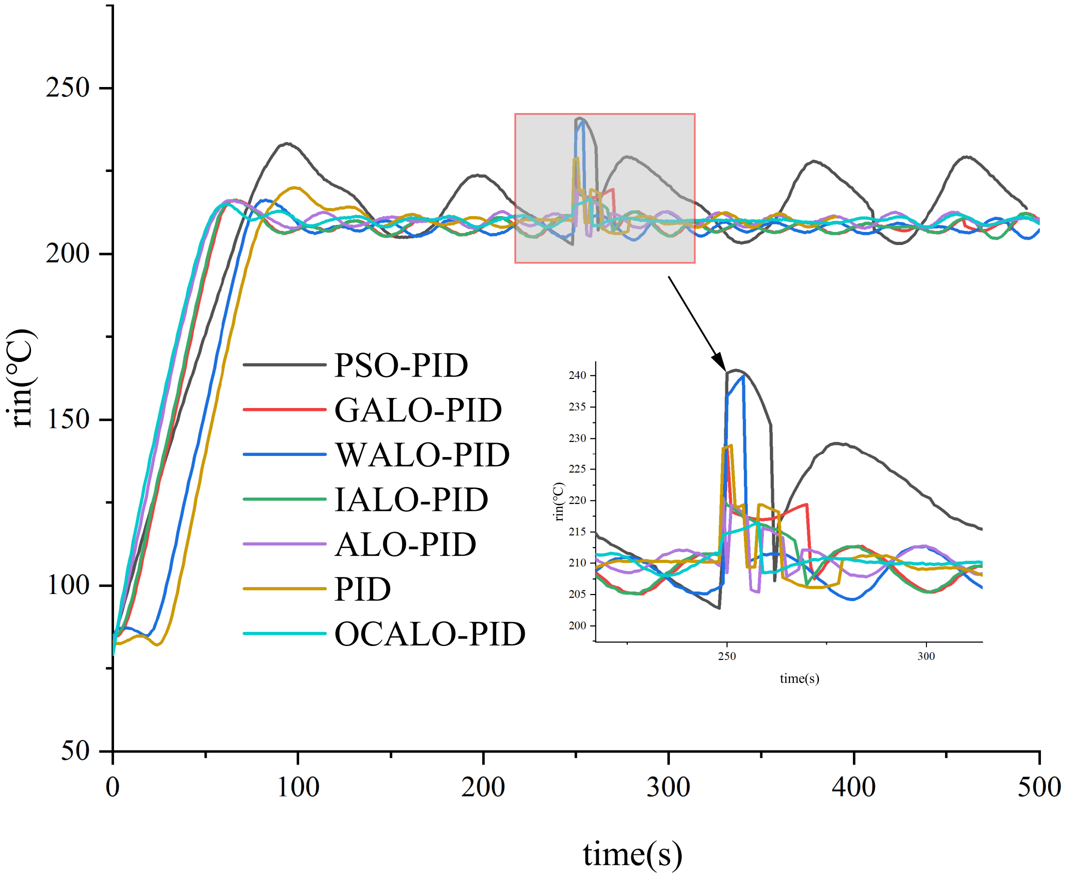


Figure 13. Perturbation test curve graph.

Table 12. Performance index.

Algorithm	Steady-State Error (◦C)	Recovery Time (s)
PSO-PID	30.4	-
GALO-PID	18.5	376
WALO-PID	26.7	369
IALO-PID	10.4	370
ALO-PID	9.3	352
PID	9.6	340
OCALO-PID	**6.5**	**261**

Through the perturbation test, the OCALO-PID controller restores the steady state after 261 s, which is 115 s, 108 s, 109 s, 91 s, 79 s, and 131.4 s less than PSO-PID and GALO-PID, respectively, The steady state value of the OCALO-PID controller is 216.5 °C, and the error between setpoint and 121 °C is only 6.5 °C. The recovery time is much shorter than that of the PSO-PID and GALO-PID. The steady-state value of the OCALO-PID controller is 216.5 °C, and the error between the setpoint and 121 °C is only 6.5 °C, which is 30.4 °C, 18.5 °C, 26.7 °C, 10.4 °C, 9.3 °C, and 9.6 °C lower than that of the PSO-PID and GALO-PID, respectively, The steady state errors of PSO-PID,WALO-PID, IALO-PID, ALO-PID and PID are smaller, respectively. Therefore, the OCALO-PID controller has excellent performance to meet the robustness of high-speed 3D printing.

6. Conclusions

This paper improves the ALO algorithm to solve the temperature control problem in high-speed 3D printing, which provides a model and algorithm reference for other similar temperature control systems. The main work of this paper is summarised as follows:

(1) A new Tent-Logistic-Cotangent composite chaotic mapping is proposed to improve the efficiency of population initialisation.The improved tent chaotic sequence is uniformly distributed between [0,1] with good traversal.

(2) Based on the traditional ALO algorithm, an algorithmic initialisation population method combining Tent-Logistic-Cotangent compositechaotic mapping and Elite Opposition-Based learning as well as an improved elitist formula are proposed to be applied to the algorithm.The improved ALO algorithm is placed in the same benchmark function test set with the other five algorithms to verify the superiority of the improved algorithm by comparing the mean, optimum, minimum, sample variance and the number of iterations.

(3)The improved algorithm is applied to a physical simulation platform based on Klipper firmware and compared with five other algorithms as well as the classical PID algorithm. Response experiments and anti-interference experiments are compared with other algorithms to verify that the improved algorithm has a clear response capability and stability.

In this study, a new Tent-Logistic-Cotangent chaotic mapping is used in the initialisation improvement of the algorithm, which improves the efficiency of population generation more effectively, but makes the algorithm relatively more complex and increases the difficulty of integration. Much of the main work in this paper has been to validate the feasibility of the algorithm, and the physical modelling process has been relatively crude, with further research allowing optimization of the control model to achieve the improvements. As we move towards 3D printing manufacturing, future research will be closer to practical needs, and future control systems for high-speed 3D printing will need to continually face difficult upgrades, so this paper's single addition of a disturbance is slightly insufficient, and future research can improve the simplicity of the algorithm as well as increase its reliability in the actual 3D printing process.

Author Contributions: Conceptualization, R.Z. and C.D.; methodology, J.J.; software, J.J.; validation, R.Z. and C.D.; formal analysis, J.J.; investigation, J.J.; resources, R.Z.; data curation, J.J.; writing—original draft preparation, S.L.; writing—review and editing, J.J.; visualization, J.J.; supervision, J.J.; project administration, C.D.; funding acquisition, R.Z.and S.L. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: The authors acknowledge the financial support of the Fundamental Research Funds for the Central Universities of China (3122022106) , the Special Funds for Science and Technology Program in Tianjin (22KPXMRC00190) and Civil Aviation University of China Postgraduate Research and Innovation Programme(2023YJSKC08002).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Peng Geng and Ji Zhao. Effects of extrusion speed and printing speed on the 3D printing stability of extruded PEEK filament. *Journal of Manufacturing Processes*, **2019**, *37*, 266-273.
2. Nazir A, Jeng J-Y. A high-speed additive manufacturing approach for achieving high printing speed and accuracy. *Proceedings of the Institution of Mechanical Engineers*, **2020**, *234(14):* 2741-2749.

3. X. Wu, X. Wang and G. He. A Fuzzy Self-tuning Temperature PID Control Algorithms for 3D Bio-printing Temperature Control System . *2020 Chinese Control And Decision Conference (CCDC)*, **2020**, *pp.* 2846-2849.
4. V. Schmidtke, M. Rüger and O. Stursberg. Model Predictive Control of PDEs for Temperature Control in 3D-Printing Processes. *2023 European Control Conference (ECC)*, **2023**, *pp.* 1-6.
5. Liu L H , Wei X K .Multiple Model Based Robust PID Control for Superheated Steam Temperature System[J]. *Applied Mechanics and Materials*, **2014**, *494-495:* 1266-1269.
6. Astrom K J , Hagglund T.The future of PID control[J]. *Control Engineering Practice*, **2001**, *9(11):* 1163-1175.
7. Seyedali,Mirjalili,Pradeep,et al.Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems[J]. *Applied Intelligence*, **2016**.
8. K, Shankar and Lakshmanaprabu S.K.Optimal key based homomorphic encryption for color image security aid of ant lion optimization algorithm. *International journal of engineering and technology*, **2022**,*2022(187-):* 187.
9. Kesarwani S , Verma R K.Ant Lion Optimizer (ALO) algorithm for machinability assessment during Milling of polymer composites modified by ze-ro-dimensional carbon nano onions (0D-CNOs)[J].*Measurement*, **2018**,*7 (2018):* 22.
10. Li,Y.; Wu,Y.Gaussian mutation based ALO and its application in combinatorial optimization[J]. *China Sciencepaper*, **2022**,*17(03):* 295-304.
11. Wang,Y. ; Yu, H.; Ling D. ; et al.Optimization of BP for bearing fault diagnosis based on improved antlion algorithm[J/OL]. *Computer Integrated Manufacturing Systems*, **2023**, 1-21.
12. Ji, J.; Wang, F.; Zhou, M.; Guo, R.; Ji, R.; Huang, H.; Zhang, J.; Nazir, M.S.; Peng, T.; Zhang, C.; et al. Evaluation Study on a Novel Structure CCHP System with a New Comprehensive Index Using Improved ALO Algorithm. *Sustainability* , **2022**, *14*, 15419.
13. Mirjalili, Seyed Mohammad. The Ant Lion Optimizer. *Adv. Eng. Softw*, **2015**, *83 (2015):* 80-98.
14. Zhongyun Hua, Yicong Zhou, Hejiao Huang.Cosine-transform-based chaotic system for image encryption. *Information Sciences*, **2019**, *480*, 403-419.
15. 16. H. R. Tizhoosh.Opposition-Based Learning: A New Scheme for Machine Intelligence. *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents*, **2005**, *pp.*695-701.
16. Zhao, Jia and Li Lv.Shuffled frog-leaping algorithm using elite opposition-based learning. *Int. J. Sens*, **2019**, *Networks 16 (2014):* 244-251.
17. Yildiz, B.S., Pholdee, N., Bureerat, S. et al. Enhanced grasshopper optimization algorithm using elite opposition-based learning for solving real-world engineering problems.*Engineering with Computers*, **2022**, *38*,4207–4219.
18. Guo, J., Chen, C., Xiang, S. et al.A fast bilateral filtering algorithm based on rising cosine function. *Neural Computing and Applications*, **2019**,1-12.
19. Saeed, S.; Ong, H.C.; Sathasivam, S.Self-Adaptive Single Objective Hybrid Algorithm for Unconstrained and Constrained Test functions: An Application of Optimization Algorithm. *Arab. J. Sci. Eng.* **2019**, *44*, 3497–3513.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.