International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 2473-2485

https://doi.org/10.70517/ijhsa463206

Optimization Problem Solving Based on Numerical Analysis Algorithms and Its Application in Artificial Intelligence

Jun Zhang[1,*] and Jie Chen[2]
[1] Department of Public Teaching, Xinyang Vocational College of Art, Xinyang, Henan, 464000, China
[2] School of Preschool Education, Xinyang Vocational College of Art, Xinyang, Henan, 464000, China
Corresponding authors: (e-mail: 13137606238@163com).

Abstract Gray wolf optimization algorithm has performance defects such as slow convergence speed, easy to fall into local optimum and high processing complexity. In this paper, an improved gray wolf optimization algorithm (IGWO) is proposed with reference to the improvement idea of whale optimization algorithm. Aiming at the shortcomings of the uneven distribution of individuals in the population initialization of the grey wolf optimization algorithm, a chaotic mapping is introduced to initialize the population, and based on the randomness and regularity of the chaotic mapping, the global search process of the algorithm in this paper is optimized. In addition, an exponential convergence factor is proposed to update the control parameters of the algorithm. Then, a nonlinear convergence parameter is introduced to change the updating method of the iterative formula to optimize the algorithm, and it is applied to artificial intelligence. The research results show that the clustering accuracy of the improved Gray Wolf Optimization Algorithm for a variety of classical functions is maintained at more than 90%, the convergence accuracy and performance are better than the comparison algorithm, and it is able to complete the multi-intelligent path planning under different experimental environments and different number of intelligences control conditions, which verifies the feasibility of the algorithm proposed in this paper in solving the optimization problem of the Gray Wolf Optimization Algorithm.

Index Terms Gray wolf optimization algorithm, chaotic mapping, nonlinear convergence parameter, multi-intelligent body path planning

I. Introduction

Numerical analysis algorithms refer to a class of methods for computation and optimization through mathematical models, which includes a variety of technical means, such as interpolation, numerical calculus, linear algebra, etc., and can be used to solve practical problems in the fields of scientific computation, engineering design and so on [1]-[4]. In practical applications, the optimization and implementation of numerical analysis algorithms are very important; optimization makes numerical methods more efficient, accurate and reliable, and implementation transforms numerical methods into computer programs for computation [5]-[7].

Artificial intelligence, as a cross-discipline, covers a variety of fields such as machine learning and deep learning [8]. When applying AI algorithms, parameter optimization in AI algorithms is the key to improve the performance and effectiveness of the algorithms [9], [10]. However, parameter tuning in AI algorithms is usually a complex and time-consuming task [11]. Artificial intelligence algorithms usually have a large parameter space and the parameters interact with each other, so an efficient and accurate optimization method is needed [12]. Numerical analysis algorithms provide a viable way to optimize and tune the parameters in AI algorithms, thus making them more accurate and efficient. In artificial intelligence, numerical analysis algorithms improve the efficiency and accuracy of numerical methods by selecting algorithms, parameter tuning, numerical error control, and parallel computing optimization.

In this paper, the improvement of Gray Wolf optimization algorithm is carried out from three aspects, namely, population initialization, convergence factor and control parameters and position updating formula, in response to the shortcomings of Gray Wolf optimization algorithm. Bernoulli chaotic mapping initialization method is first used to improve the coverage of population individuals in the solution space to enrich the diversity of the population. Then the convergence factor of the algorithm is improved by an exponential convergence factor updating strategy to improve the convergence accuracy of the algorithm. The dynamic weighting factor and the adaptation scale factor, which are linearly decreasing, are introduced to focus on the head wolf with the best objective function value according to the different adaptation values, so that the algorithm jumps out of the local optimal solution. Multi-intelligent body coordination strategy is added to the improved algorithm so that it can avoid collision between multi-intelligent bodies. The effectiveness of the improved gray wolf optimization algorithm is verified through comparative simulation experiments with classical test functions, and the algorithm is applied to the simulation of optimal path planning of multiple intelligences in different experimental environments.

II. Optimization problem solving based on grey wolf optimization algorithm

II. A. Mathematical model of the gray wolf optimization algorithm

II. A. 1) Surrounding prey

Gray wolves encircle prey during hunting. Mathematical modeling of the encircling behavior presents the following equation:

$$D = | C \cdot X_p(t) - X(t) | \tag{1}$$

$$X(t+1) = X_p(t) - A \cdot D \tag{2}$$

where t denotes the current iteration number, A and C are coefficient vectors, $X_p(t)$ denotes the position vector of the prey, $X(t)$ denotes the current position vector of the gray wolf, D denotes the current distance between the gray wolf and the prey, and $X(t+1)$ denotes the update of the gray wolf position.

The coefficient vectors A and C are formulated as follows:

$$A = 2ar_1 - a \tag{3}$$

$$C = 2r_2 \tag{4}$$

where r_1, r_2 are random vectors in [0, 1], and a is a convergence factor affecting the change of A, which decreases linearly from 2 to 0 with the number of iterations, with the following expression:

$$a = 2(1 - t / t_{\max}) \tag{5}$$

where $t_{\max}$ denotes the maximum number of iterations.

II. A. 2) Chasing prey

The prey is the target minimum, and α is the value closest to the optimal solution, followed by β and δ, and the rest of the scenarios are ω, where ω does not represent just one.

In the hunting process, the different positions of the three wolves of α, β, δ are used to predict the position of the prey, and the mathematical model is as follows:

$$\begin{cases} D_\alpha = | C_1 o X_\alpha - X | \\ D_\beta = | C_2 o X_\beta - X | \\ D_\delta = | C_3 o X_\delta - X | \end{cases} \tag{6}$$

where $D_\alpha, D_\beta, D_\delta$ represent the approximate distances between the current gray wolves converging to the α, β, δ wolves, respectively; C_1, C_2, C_3 are coefficient vectors, which denote the search ranges of the three wolves.

Each wolf updates its position as follows:

$$\begin{cases} X_1 = X_\alpha - A_1 \cdot D_\alpha \\ X_2 = X_\beta - A_2 \cdot D_\beta \\ X_3 = X_\delta - A_3 \cdot D_\delta \end{cases} \tag{7}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{8}$$

where $X_\alpha, X_\beta, X_\delta$ denotes the current position of the α, β, δ wolf, and X_1, X_2, X_3 denotes the length and direction of steps taken by the ω wolf toward α, β, δ the step length and direction of advancement. $X(t+1)$ denotes the final position of the ω wolf.

II. A. 3) Attacking prey

In the GWO algorithm [13], the balance between exploration and exploitation is mainly determined by parameter A and parameter C. When $|A| > 1$, the GWO algorithm forces the search operator to search the prey and emphasizes global exploration; when $|A|<1$, the GWO algorithm focuses more on fast convergence towards the prey and emphasizes local exploitation.

Meanwhile another thing that helps search in GWO is C, C is a random value between [0, 2], the vector provides random weights for the prey as a way to emphasize or weaken the effect of the prey in defining the distance in the equation. C helps the gray wolf to be more random throughout its search for an optimum, avoiding falling into a local optimum while searching more comprehensively. C gives a random weight value to the prey, and strengthening $C(C>1)$ makes it increasingly difficult for the wolf to approach the prey; conversely weakening $C(C<1)$ enables the wolf to approach the prey more quickly.

II. B. Basic Steps and Flowchart of Gray Wolf Optimization Algorithm

The steps of the GWO algorithm are summarized as follows:

Step 1: Set the number of population sizes N, the maximum number of iterations $t_{\max}$, the search dimension D; generate the parameters a, A and C to randomly initialize the gray wolf population.

Step 2: Calculate the fitness values of all individual gray wolves in the population and rank them, record the fitness of the top three rankings and record the positions as X_α, X_β and X_δ.

Step 3: Update the convergence factor a and the values of the coefficient vectors A and C.

Step 4: Update the position of each individual of the population.

Step 5: Calculate the fitness of each individual and update the fitness and position of α, β, δ wolf.

Step 6: Judge whether the end condition is satisfied, if the predetermined maximum number of iterations $t_{\max}$ is reached, then stop the calculation, otherwise repeat the loop to jump to step 2.

Step 7: Output the global optimal solution.

The specific flowchart of the GWO algorithm is shown in Fig. 1.

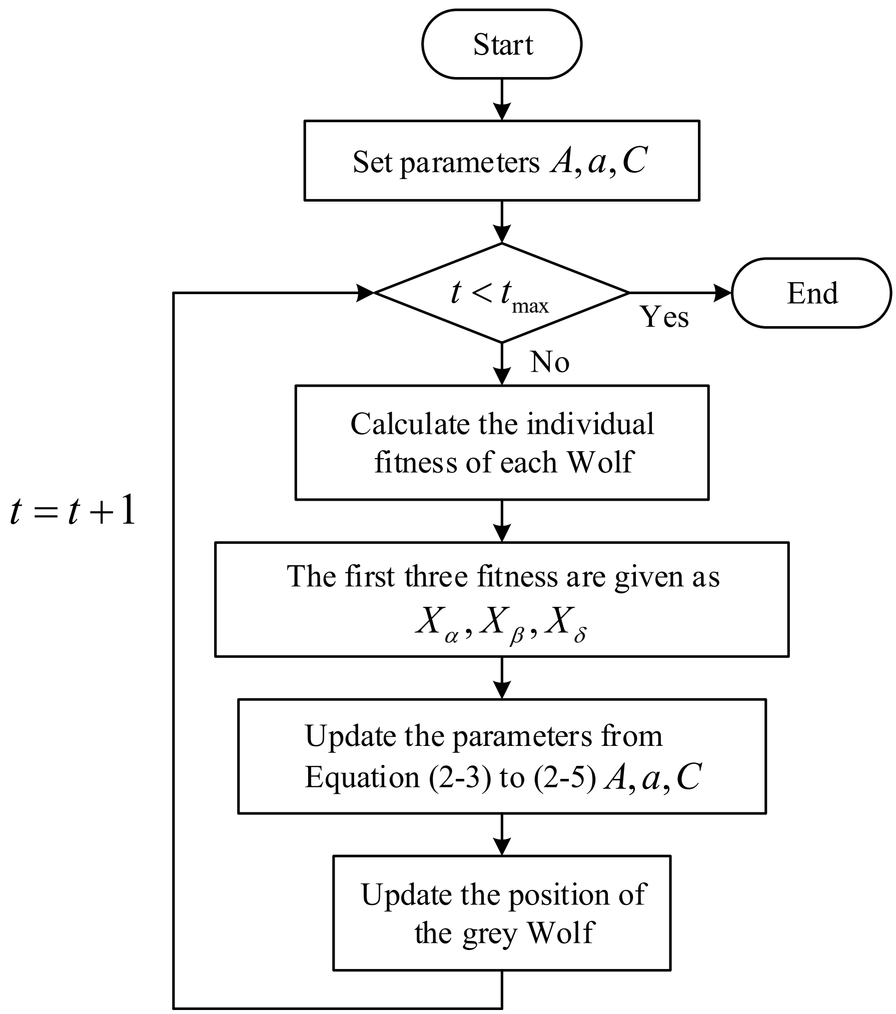


Figure 1: Flowchart of GWO algorithm

II. C. Improvement of Gray Wolf Optimization Algorithm

Although the Gray Wolf optimization algorithm has the advantages of simple structure and easy implementation, the algorithm also has some obvious shortcomings, such as the lack of diversity in the population, imbalance between the exploitation and exploration phases, and premature convergence during the iteration process. This is similar to the characteristics of another swarm intelligence optimization algorithm, Whale Optimization Algorithm (WOA) [14], and there have been a number of studies on the improvement of WOA over the years. Using the improved whale algorithm to simulate and optimize the cold chain distribution model, the traditional whale algorithm is easy to fall into local extremes and convergence speed problems are solved by initializing the sobol sequence for the random number cluster of individual whales and introducing the golden sine algorithm to update the position formula of searching individuals. The whale optimization algorithm is improved by introducing nonlinear convergence parameters and changing the updating method of iterative formulas, and the improved whale optimization algorithm (IWOA) [15] is used to rectify the PID parameters, which achieves good results.

II. C. 1) Chaotic initialization of algorithmic populations

Since the population initialization of the original gray wolf algorithm is randomly generated, this makes the distribution of individuals extremely uneven, which cannot reflect the diversity of the population and affects the algorithm's optimization search. While the typical characteristics of chaotic mapping are randomness, ergodicity, regularity, etc., which can ensure the diversity of the population in order to optimize the global search process. Therefore, this study adopts the method of Bernoulli chaotic mapping to improve the population initialization in order to enhance the population diversity of the gray wolf algorithm. The mathematical model formulas are shown in (9) and (10).

$$Z_{k+1}=\begin{cases}\dfrac{Z_k}{(1-\lambda)}, & Z_k\in(0,1-\lambda]\\ \dfrac{(Z_k-1+\lambda)}{\lambda}, & Z_k\in(1-\lambda,1)\end{cases} \tag{9}$$

$$X_k=X_k^{\max}+Z_k(X_k^{\max}-X_k^{\min}) \tag{10}$$

In the above equation, k is the population size, Z_k is the generated chaotic sequence, and the value of λ is a random number between 0 and 1.

The chaotic sequence Z_k is then combined to further generate a sequence of initial locations X_k of individual gray wolves in the search area.

II. C. 2) Improvement of convergence factors and control parameters

In the mathematical model of the gray wolf algorithm, the coefficient vectors $\vec{A},\vec{C}$ are the key parameters controlling the search range of the wolf pack, in which $\vec{A}$ denotes the search radius of the wolf pack, which is used for adjusting the spacing between the wolf pack and the prey in phases, and at the same time, the control parameter $\vec{C}$ also coordinates the ability of the gray wolf algorithm for the global exploration and local exploitation. And these two parameters are related to the convergence factor a and the random vectors $r1$, $r2$, so this study improves the control parameters by proposing an exponential convergence factor a updating strategy, which can better fit the actual nonlinear change process of the convergence factor a, and the equations are shown in (11) and (12).

$$a(l)=2-\sqrt{2}\times((e^{\frac{l}{maxiter}}-1)^{\lambda_1})^{\lambda_2} \tag{11}$$

$$\vec{C}=2\times r_3-a \tag{12}$$

In the above equation, l is the number of iterations, maxiter is the maximum number of iterations, λ_1,λ_2 are random numbers between 1-6, and r_3 are random numbers.

II. C. 3) Improvement of the position update formulae

In order to better develop the search-optimization capability of the gray wolf algorithm, weigh the different guiding effects of the best three wolves on the positional updates of the remaining individual gray wolves, and prevent falling into a localized range of premature stagnation, firstly, a dynamic weight factor b that varies in a linearly decreasing manner is introduced, and secondly, a fitness scaling factor is introduced v_1,v_2,v_3. The formulas are shown in (13) to (15).

$$b(l) = b_f - \frac{l}{maxiter} \times (b_f - b_s) \tag{13}$$

$$\begin{cases} f = \left| f_\alpha + f_\beta + f_\delta \right| \\ v_1 = \dfrac{f_\alpha}{f}, v_2 = \dfrac{f_\beta}{f}, v_3 = \dfrac{f_\delta}{f} \quad f > 0 \\ v_1 = v_2 = v_3 = \dfrac{1}{3} \quad f = 0 \end{cases} \tag{14}$$

$$X(l+1) = b(l) \times r_4 \times (v_1 \times X_1 + v_2 \times X_2 + v_3 \times X_3) \tag{15}$$

In the above equation, b_s and b_f denote the initial and final values of the weighting factors in turn, $f_\alpha, f_\beta, f_\delta$ denote the values of the three kinds of wolf adaptations, respectively, and r_4 is a random number.

In summary, the overall flowchart of the improved gray wolf optimization algorithm can be obtained as shown in Fig. 2.

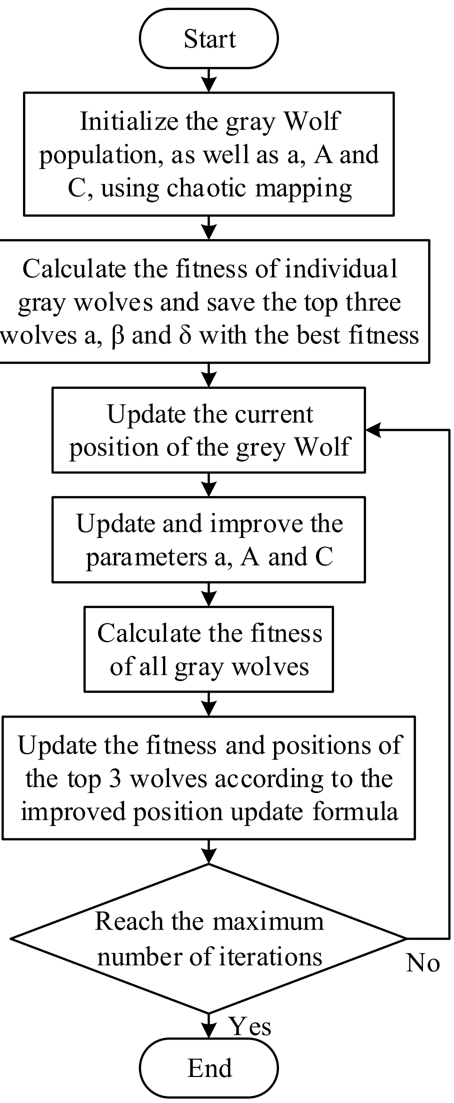


Figure 2: Flowchart of the improved Grey Wolf algorithm

II. D. Classical test function simulation of the improved algorithm

The single-objective function can be regarded as a special case of multi-objective function, and f_1: Sphere function f_2, Ackley function f_3, Griewank function f_4, and Rastrigrin function are selected for experimental simulation. The parameters are set as follows $Q=350, T_{\max}=400, C_{\min}=5C_{\max}=50, H=8, \omega_1=0.35, \omega_2=0.45, \omega_3=0.20$.

Multi-metric adaptive FCM performance validation is carried out using the self-clustering FCM algorithm proposed in the literature and the randomized fuzzy clustering algorithm in the literature for comparative experiments, and populations with different iterations are selected for cluster analysis, and the evaluation indexes are set as clustering validity index V_D, and clustering accuracy Ω:

$$V_D=(\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}^{m}\|v_i-x_j\|^2)/(N\min_{i\neq j}\|v_i-v_j\|^2) \tag{16}$$

$$\begin{aligned}\Omega=&\sum_{j=1}^{N}((\sum_{x_j\in M_j}\|x_j-x_z\|)\sqrt{(N\times|M_j|)})\\&-\sum_{j=1}^{N}(\sum_{x_j\in Ne}\|x_j-x_z\|)(N\times|N_j|)\end{aligned} \tag{17}$$

where: M_j a set of data that is dissimilar to x_j; E_j a set of data that is similar to x_j.

The smaller the value of V_D the better the clustering result. The performance comparison of different clustering algorithms is shown in Table 1.

For high-dimensional complex function f_2, pathological complex function f_4, both the clustering validity index V_D and clustering accuracy Ω, the clustering algorithms here are better than the other two algorithms, in particular, for the pathological complex function f_4, the clustering accuracy of the algorithms here is above 90%, different degrees than the other algorithms. That are optimized to different degrees than the other algorithms. This shows that the introduction of multimetric high-dimensional mapping functions effectively improves the clustering ability of the clustering algorithm for complex data, and the clustering effect is better.

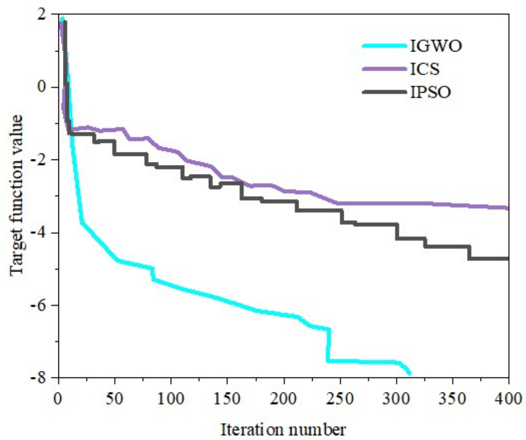
Table 1: Comparison of performance of different clustering algorithms

Evaluation index		f2			f4		
		t=20	t=100	t=200	t=20	t=100	t=200
C	This algorithm	17	10	6	22	14	11
	Self-clustering FCM algorithm	27	17	14	25	14	16
	Random fuzzy clustering algorithm	22	14	7	27	14	14
VD	This algorithm	0.22	0.16	0.16	0.33	0.26	0.15
	Self-clustering FCM algorithm	0.5	0.41	0.44	0.65	0.56	0.49
	Random fuzzy clustering algorithm	0.5	0.4	0.34	0.66	0.72	0.48
Ω	This algorithm	95.23	96.8	96.72	94.48	93.77	95.63
	Self-clustering FCM algorithm	85.12	86.31	82.39	65.19	62.37	61.24
	Random fuzzy clustering algorithm	64.33	77.23	80.68	79.85	74.29	81.91

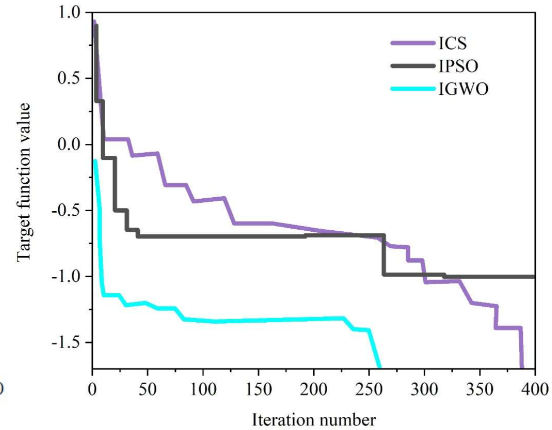
Different intelligent optimization algorithms are selected for comparison experiments with improved cuckoo optimization algorithm (ICS) and improved particle swarm optimization algorithm (IPSO). The evaluation indexes are set as the maximum value $Y_{\max}$, the minimum value $Y_{\min}$, the mean value $\overline{Y}$ and the operation time Y_T. The convergence curves of the functions are shown in Fig. 3, (a) to (d) are the convergence curves of Sphere function, Ackley function, Griewank function and Rastrigrin function, respectively. The evaluation index comparison results are shown in Table 2.

In terms of convergence accuracy, for the IGWO and IPSO algorithms, both algorithms are able to find the optimal solutions of the three functions in the range of convergence accuracy, but the convergence accuracy of IGWO is significantly better than that of IPSO. For the ICS algorithm, it is able to get the global optimal solutions of f1, f2, and f3, but it is not able to get the global optimal solution of f4, and its convergence accuracy is poorer than that of the other two algorithms. In terms of operation time, the convergence time is longer than the other two algorithms because IGWO iteratively performs cluster analysis operations. Comparison experiments of different intelligent optimization algorithms show that the global convergence accuracy of the algorithm is effectively improved by the chaotic initialization of the algorithm population, the improvement of the convergence factor and control parameters, and the improvement of the position updating formula, and the result of the search for the optimum is better.The

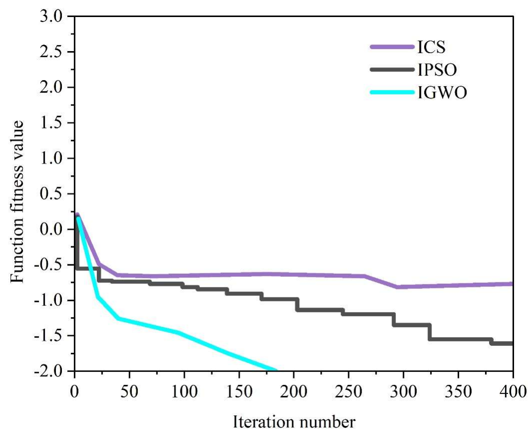
Ymax, Ymin, and Ymean indexes of the IGWO algorithm are always the smallest, and the Ymin is only 0 in the f4 function.



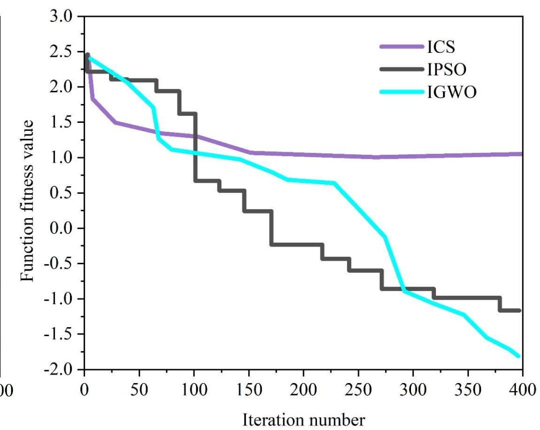
(a) Sphere function convergence curve



(b) Ackley function convergence curve



(c) Griewank function convergence curve



(d) Rastrigrin function convergence curve

Figure 3: Function convergence curve

Table 2: Comparison of Evaluation Indexes

Function	Algorithm	Index			
		Ymax	Ymin	Ymean	YT
f1	IGWO	2.77e-8	9.91e-8	4.12e-8	11.65
	IPSO	7.84e-4	2.15e-4	3.36e-4	16.29
	ICS	3.21e-2	1.15e-2	3.75e-3	23.09
f2	IGWO	2.08e-8	1.02e-7	1.15e-8	35.22
	IPSO	5.32e-4	7.27e-6	2.36e-5	15.7
	ICS	0.14	1.24e-2	0.56	11.83
f3	IGWO	5.77e-7	2.02e-7	3.55e-8	24.17
	IPSO	6.63e-5	8.72e-6	3.23e-5	15.37
	ICS	2.35e-3	1.12e-4	6.27e-2	9.9
f4	IGWO	3.32e-8	0	8.04e-8	20.07
	IPSO	1.58e-6	2.14e-7	2.86e-5	10.23
	ICS	0.26	0.07	0.13	9.29

III. Multi-intelligent body path planning

III. A. Multi-intelligent body path planning problem description

The purpose of multi-intelligent body path planning [16] is as follows:

(1) To find the optimal path without collision with obstacles for each intelligent body to move from the current position to the target position through an algorithm;

(2) To design a coordination strategy between the intelligences to prevent collisions between the intelligences;

(3) The final found multi-intelligent body movement path should be in line with the anthropomorphic characteristics of intelligent body movement, i.e., the path traveled by each intelligent body should be as smooth as possible, and the turning amplitude should be small.

III. A. 1) Path coding

For the multi-intelligent body path planning problem, this chapter adopts the path point sequence encoding method to simplify the two-dimensional path point coordinates in the plane coordinate axes into one-dimensional y axis coordinates. Unlike the single-intelligent body path planning encoding, it is necessary to set the values of the transverse coordinates of each intelligent body in the same planar coordinates, i.e., the transverse coordinates of Intelligent Body 1, $x_{1,1}, x_{1,2}, \cdots x_{1,n}$, and the transverse coordinates of Intelligent Body 2, $x_{2,1}, x_{2,2}, \cdots x_{2,n}$ are different values. As an example, the initial points of two intelligences are $Y_{1,0}(x_{1,0}, y_{1,0}), Y_{2,0}(x_{2,0}, y_{2,0})$ and the termination points are $Y_{1,3}(x_{1,3}, y_{1,3})$, $Y_{2,3}(x_{2,3}, y_{2,3})$, and the intermediate path points are on the perpendicular lines of the transverse coordinate points $x_{1,1}, x_{1,2}$ and $x_{2,1}, x_{2,2}$.

The intermediate waypoints of each agent are determined as follows: the initial point $Y_{i,0}(x_{i,0}, y_{i,0})$ and the end point $Y_{i,n+1}(x_{i,n+1}, y_{i,n+1})$ are equally divided into $n+2$ perpendicular lines of the x axis, since $x_{i,0}$ and $x_{i,n+1}$ are the first i The abscissa values of the initial and end points of the agent in the direction of the x axis, $x_{i,1}, x_{i,2}, \cdots x_{i,n}$ divide $x_{i,0}$ and $x_{i,n+1} n+1$ with respect to the x axis, so any abscissa point $x_{i,j} = x_{i,0} + j \cdot (x_{i,n+1} - x_{i,0}) / (n+1)$ is known, and only the corresponding y axis coordinates $y_{i,1}, y_{i,2}, \cdots y_{i,n}$ can be determined to determine the exact position of the i agent at any waypoint. A robot path is formed by connecting $n+2$ coordinate points including the initial point and the goal point. Where n coordinate points $y_{i,1}, y_{i,2}, \cdots y_{i,n}$ form an individual, each individual represents a path, and all paths are summed up to get the total path of all intelligences. In the evolutionary process, the optimal individual obtained is the shortest path without passing through obstacles.

III. A. 2) Optimization objectives and constraints for mathematical models

(1) Optimization Objective

For the multi-intelligent body path planning problem, design the path model that makes the multi-intelligent body from the respective starting point to the goal point. In this chapter, each intelligent body is reduced to a prime point, and each obstacle is set as a circle with different radii, so that the shortest path of multiple intelligent bodies is optimized as the optimization objective.

(2) Constraints

1) Handling of infeasible paths and infeasible path points: if an intelligent body collides with an obstacle, this intelligent body path needs to be corrected;

2) Smoothness: the path traveled by each intelligent body should be as smooth as possible, in line with the anthropomorphic characteristics of intelligent body movement.

The mathematical model of multi-intelligent body operation is as follows:

$$f = \min \sum_{i=1}^{m} \sum_{j=1}^{n+1} \omega_{i,j} \sqrt{(y_{i,j} - y_{i,j-1})^2 + b_i^2} \tag{18}$$

$$\omega_{i,j} = \begin{cases} \varepsilon & (S_{i,kj} \le S_{dk}) cap (S_{i,k} = 0) \\ 1 & \text{other} \end{cases} \tag{19}$$

$$S_{i,kj} = \frac{|(y_{i,j} - y_{i,j-1})(x_{i,j} - x_{i,j-1}) / b_i - y_k + y_{i,j}|}{\sqrt{((y_{i,j} - y_{i,j-1}) / b_i)^2 + 1}} \tag{20}$$

$$S_{dk} = R_k + \delta \tag{21}$$

$$b_i = (x_{i,n+1} - x_{i,0}) / (n+1) \tag{22}$$

In the above equation, f denotes the sum of the shortest paths run by m intelligences (where each intelligence has n+1 path points). Let the start position coordinates of the i th intelligent be $(x_{i,0}, y_{i,0})$, the target coordinates be

$(x_{i,n+1}, y_{i,n+1})$, and $x_{i,1}, x_{i,2}, \dots x_{i,n}$ are the shortest paths to be taken by combining $x_{i,0}$ and $x_{i,n+1}$ after equidistributing $x_{i,n+1}$ with respect to the x-axis n+1, and b_i is the equidistribution between $x_{i,0}$ and $x_{i,n+1}$. The $y_i, y_{i,2}, \cdots y_{i,n}$ are the path points corresponding to $x_{i,1}, x_{i,2}, \cdots x_{i,n}$; taking into account the fact that the line connecting the two neighboring path points may intersect with the obstacles and form an infeasible path, the penalization function $\omega_{i,j}$ to handle this situation. The $\varepsilon > 1$ is the penalty factor, which can make the infeasible path longer; $S_{i,kj}$ denotes the perpendicular distance from the center of the circle of the obstacle $k(x_k, y_k)$ to the straight line where the two neighboring path points $y_{i,j} y_{i,j-1}$ are located; and S_{dk} denotes the range of influence of the k th obstacle, R_k is the radius of the k th obstacle, and $\delta = rand(0, 0.1)$ is a smaller number; $S_{i,k}$ denotes whether or not the line segment $y_{i,j} y_{i,j-1}$ intersects with the k th obstacle, and intersection $S_{i,k} = 0$, otherwise $S_{i,k} = 1$.

III. B. Application of IGWO algorithm to multi-intelligent body path planning

For the multi-intelligent body system obstacle avoidance problem, the IGWO algorithm is used to solve the problem, and the adaptive operator is employed to obtain a better convergence speed as well as a stronger global search ability, meanwhile, for the different intelligences, the individuals are divided into different sub-populations, and the matching variance operator is introduced accordingly, which converts the multi-intelligent body path problem into a parallel optimization problem with multiple populations.

III. B. 1) Population initialization

In initializing the population, let D be the dimension of the population and NP be the size of the population, the population initialization process is shown in Fig. 4, where the initial population is divided into n sub-populations, each of which plans a path to an intelligent body. Each sub-population has m individuals, and the individuals in each sub-population are generated in their respective solution spaces. The relationship between sub-populations and population size is $n * m = NP$.

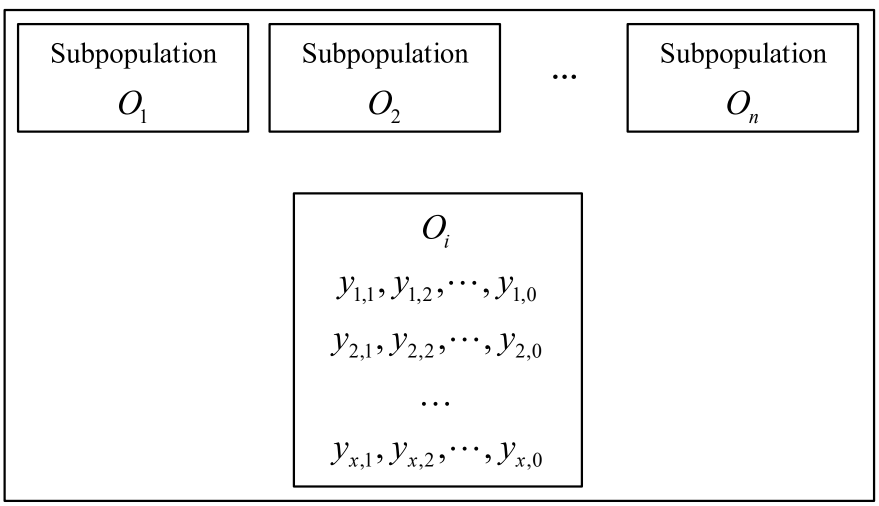


Figure 4: Multi-population representation

When the initial and target points of each agent are $(x_{1,0}, y_{1.0}), (x_{2,0}, y_{2.0}), \cdots, (x_{n,0}, y_{n.0})$ and $(x_{1,D+1}, y_{1,D+1}), (x_{2,D+1}, y_{2,D+1}), \cdots, (x_{n,D+1}, y_{n,D+1})$, the selection method of the initial individual of the algorithm is as follows,

Calculate the length of the start point and the target point about the x-axis for each intelligence as $L_i = | x_{i,n+1} - x_{i,0} |$, respectively, to get the largest L_i, and then compare the sizes of all the $y_{i,0}$ and $y_{i,n+1}$, to get the one with a larger value as $y_{\max}$ and the smaller value is $y_{\min}$. Then the upper bound of the population of the algorithm is $y_{\max} + L/2$ and the lower bound of the population is $y_{\min} - L/2$. After determining $x_i, x_{i,2}, \dots x_{i,n}$, nodes can be selected for each subpopulation to form path points.

III. B. 2) Multi-intelligent body coordination strategy design

The multi-intelligent body coordination strategy in this chapter adopts the geometric modification method, which coordinates the paths between intelligences by changing the geometric paths in the region where the collision of intelligences occurs. The initial points of the two intelligences are $y_{1,0}, y_{2,0}$, and the target points are $y_{1,5}, y_{2,5}$, respectively. The path points $y_{1,3}$ and $y_{2,3}$ intersect, and at this point it is necessary to determine whether Intelligent Body 1 and Intelligent Body 2 collide. Let the two intelligences move at a constant speed of v_1 and v_2

respectively, the distance from the initial point to the intersection point of each intelligent body can be obtained, so whether a collision between intelligences occurs or not can be determined by calculating the running time of each intelligent body.

If the two intelligences run from the initial point to the intersection point in the same time, it means that the collision between the intelligences occurs, at this time you need to adjust the position of the intersection point $y_{1,3}$ and $y_{2,3}$, where $\alpha = rand(0,1)$ is the correction value, to get the intersection point $y_{1,3}$ and $y_{2,3}$ for the modified path points $y_{1,3}^{'}$ and $y_{2,3}^{'}$.

$$y_{i,j}^{'} = y_{i,j} + \alpha \cdot (y_{i,0} + y_{i,n+1} - 2 \cdot y_{i,j}) \tag{23}$$

Intelligent body paths are replanned by comparing the size of the modified path lengths $S_1^{'}, S_2^{'}$ of Intelligent Body 1 and Intelligent Body 2 with the original path lengths S_1, S_2, if $(S_1^{'} - S_1) > (S_2^{'} - S_2)$, then intelligent body 2 adopts the modified path and intelligent body 1 adopts the original path, otherwise the opposite.

III. B. 3) Algorithmic steps

Step1: Parameter design and initialization of DE population, set the population size and dimension of each sub-population of DE;

Step2: each sub-population carries out the variation, crossover and infeasible path point correction strategies of DE in Chapter 3 to obtain the optimal path of each sub-population;

Step3: Convergence operation is performed to obtain the best fitness value of individuals in each subpopulation on the local bulletin board;

Step4: Perform the coordination strategy to correct the path points that are in conflict;

Step5: determine whether the algorithm can be terminated or not, if the algorithm can be terminated then end, otherwise continue;

Step6: add 1 to the algebra and go to Step2.

IV. Simulation of IGWO algorithm for multi-intelligent body path planning applications

IV. A. Parameterization

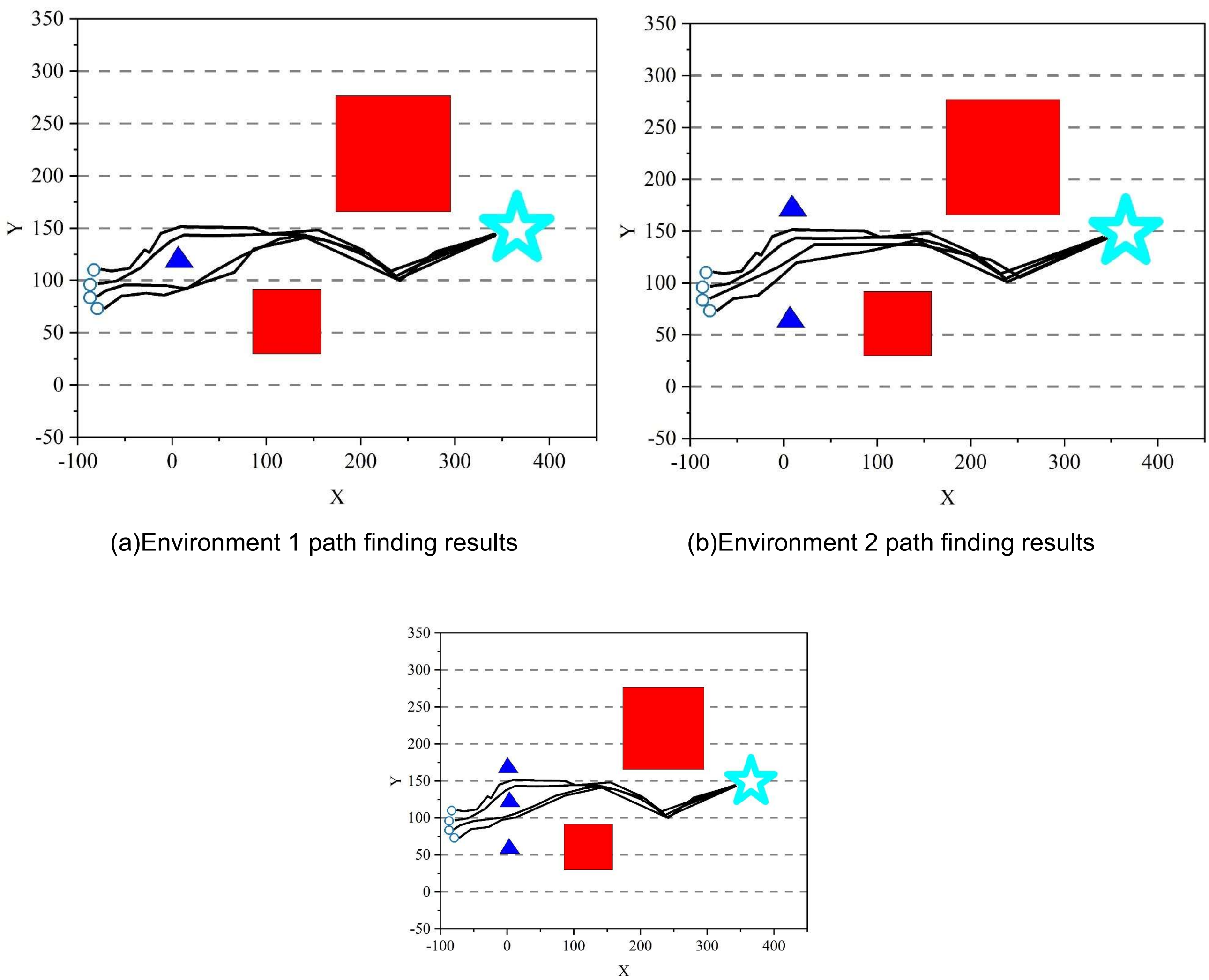
In order to validate the proposed multi-intelligent body path finding algorithm as well as the improved multi-objective gray wolf algorithm, a series of simulation environments are set up and multi-intelligent body path finding experiments are conducted in these environments. In the experimental environments, the right side of a destination location is set up to accommodate multiple intelligences, which is represented using cyan stars, and when an intelligence is less than 15 away from the destination, it is treated as if it reaches the goal. Intelligent bodies are represented using small blue circles, the initial position is near the left side of the experimental environment, and some areas that are inaccessible to the intelligent bodies are placed between the initial position of the intelligent bodies and the location of the destination, representing buildings or deep pits that may exist in reality, which are represented by red squares here for the convenience of the experiment. The algorithm in this paper will generate a shortest traveling route according to these current environmental factors. In order to efficiently conduct the experiment and verify the algorithm's obstacle avoidance ability, after generating the route, blue triangles are placed on the route to represent obstacles. Because there is a distance limitation on the detection of obstacles by the smart body, three locations are selected from the starting position of the smart body to place obstacles from near to far, and the number of obstacles in each place is also set to be different, 1, 2 or 3. A total of 3 such simulation environments were set up. 4 intelligences are set up in this set of experiments. For the optimization algorithm used in the experiment, the number of populations is set to 20 and the number of iterations is set to 50.

For the intelligences, it is specified some parameters indicating the image speed of the intelligences (8<V<16), the direction angle (ψ<0.2Pi), the detection distance 60, the detection angle 0.5 pi, the minimum safe distance from each other 2, the safe distance from the obstacles 3, and the communication distance 30. The desired safe distance between the intelligences 15. If during the run of the intelligences towards the target, there is a collision, the state of that intelligent body will change to termination and will no longer participate in the path planning task, while the other intelligences can continue the path planning.

IV. B. Path finding experiments in different environments

In order to investigate the robustness of the designed algorithms, path planning experiments in different environments are carried out. A set of initial positions are first set up, and then multi-intelligent body path finding experiments are carried out in different environments using different optimization algorithms sequentially. The results are as follows:

Figure 5 shows the roadmap obtained using the improved multi-objective gray wolf optimization algorithm. Each subgraph corresponds to an experimental environment, and Figures (a)-(c) correspond to the path finding results from environment 1 to environment 3 in turn. The four black solid lines in the figure are the movement trajectories of the four intelligences in the experiment, respectively. It can also be seen in the roadmap that the intelligent bodies will actively change direction and bypass the obstacles before they are about to encounter them, and the consistency of the route changes accordingly, indicating that the intelligent bodies enter the danger mode, at this time, maintaining the ideal formation is no longer the optimization goal of the intelligent bodies, and the intelligent bodies don't aim at maintaining a rational distance from each other, as long as they don't collide with each other. This avoidance behavior is achieved for all 3 obstacles at different distances from the initial position. When the obstacles are two or one, the individual intelligences are also able to safely pass through the gaps between the obstacles.



(a)Environment 1 path finding results

(b)Environment 2 path finding results

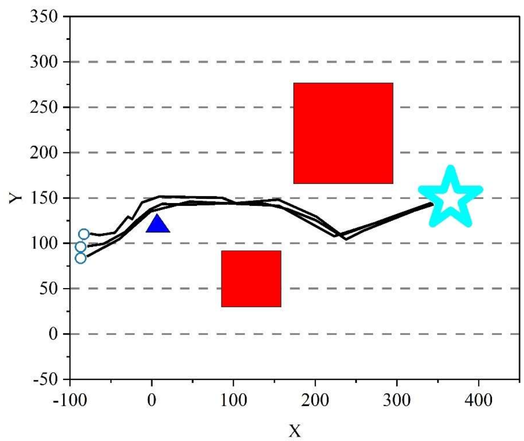
(c)Environment 3 path finding results

Figure 5: Different environmental path finding results

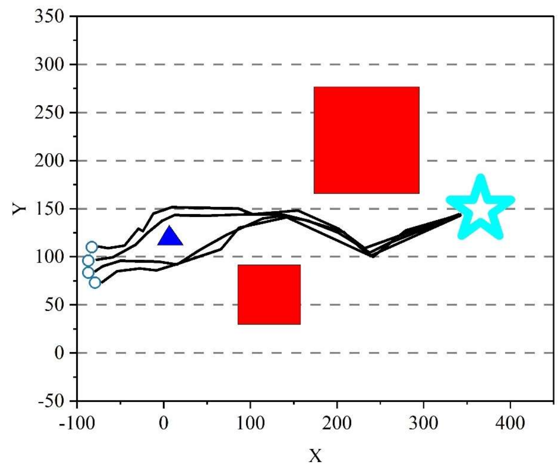
IV. C. Path finding experiments with different number of intelligences

In order to investigate the effect of the improved multi-objective gray wolf optimization algorithm on path planning with different numbers of intelligences, we conducted a set of experiments with different numbers of intelligences in Environment 1, with the number of intelligences set to increase from 3 to 5. Fig. 6 shows the roadmap obtained using the improved multi-objective gray wolf optimization algorithm. Figures (a)-(c) correspond sequentially to the path finding results for the number of intelligences from 3 to 5.

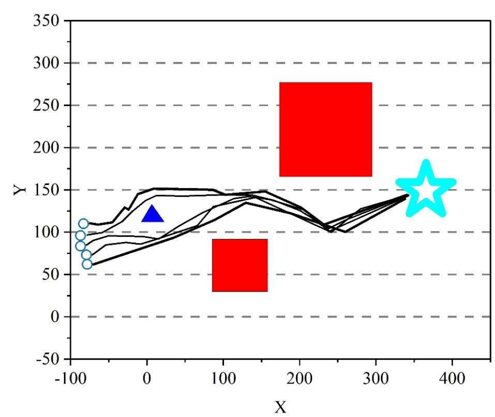
It can be seen that in all 3 experiments, the intelligent body reaches the goal location successfully. As the number of intelligences increases, the denseness of the routes generated by the intelligences increases. When passing near an obstacle, when the number of intelligences is small, they are able to pass from one side in pairs. When the number of intelligences was higher than four, they tended to split into two parts and pass from each side of the obstacle.



(a)The number of intelligent bodies is the path of 3



(b)The number of intelligent bodies is the path of 4



(c)The number of intelligent bodies is the path of 5

Figure 6: The route generated by different Numbers of intelligent experiments

V. Conclusion

In this paper, an improved gray wolf optimization algorithm (IGWO) is proposed and applied to multi-intelligentsia path planning.

(1) The clustering accuracy of the improved gray wolf optimization algorithm for different functions is kept above 90%, and the convergence accuracy is always better than that of IPSO algorithm and ICS algorithm. The Ymin of the improved Gray Wolf optimization algorithm is only 0 in the f4 function, and the Ymax, Ymin, and Ymean metrics of this paper's algorithm are smaller than those of the IPSO algorithm and the ICS algorithm, while the YT value is always better than that of the comparison algorithm. The improved Gray Wolf optimization algorithm improves the convergence accuracy and overall performance of the Gray Wolf optimization algorithm through various improvements.

(2) For the multi-intelligent body path planning problem, the improved gray wolf optimization algorithm in this paper plans the optimal path of multiple intelligent bodies respectively, and the corresponding fitness function is selected according to the different experimental environments, and the algorithm in this paper is able to safely pass through the gaps between different numbers of obstacles, plus achieve accurate path planning. When the number

of intelligent bodies increases, the route consistency generated by the intelligent bodies increases accordingly. When the number of intelligences is small, the intelligences usually pass together from the same side. When the number of intelligent bodies is large, the intelligent bodies are more inclined to split into two parts and bypass from both sides of the obstacle separately to realize the path planning.

The above results illustrate that the improved gray wolf optimization algorithm in this paper can achieve collaboration among multiple intelligences and complete the co-evolution of multiple intelligences, and the simulation study proves the practicality of this paper's method in the task of multi-intelligent body path planning.

Funding

This work was supported by Key Project of the Curriculum Reform Research Project of Teacher Education in Henan Province in 2024, "Research on the Path of Integrating Micro-certification into the Individualized Professional Development of Kindergarten Teachers" (2024-JSJYZD-074).

References

[1] Leader, J. J. (2022). Numerical analysis and scientific computation. Chapman and Hall/CRC.

[2] Li, B., & Chen, X. (2014). Wavelet-based numerical analysis: A review and classification. Finite Elements in Analysis and Design, 81, 14-31.

[3] Baskar, S., & Ganesh, S. S. (2016). Introduction to Numerical Analysis. Powai, Mumbai, India: Depart. Math., Indian Inst. Tech. Bombay.

[4] Petrović, M. J., Stanimirović, P. S., & Milovanović, G. V. (2025). Numerical Analysis and Optimization. Axioms, 14(4), 305.

[5] Song, Y., Wang, F., & Chen, X. (2019). An improved genetic algorithm for numerical function optimization. Applied Intelligence, 49, 1880-1902.

[6] Upadhyay, B. D., Sonigra, S. S., & Daxini, S. D. (2021). Numerical analysis perspective in structural shape optimization: A review post 2000. Advances in Engineering Software, 155, 102992.

[7] Jiang, H., Wang, J., Wu, J., & Geng, W. (2017). Comparison of numerical methods and metaheuristic optimization algorithms for estimating parameters for wind energy potential assessment in low wind regions. Renewable and Sustainable Energy Reviews, 69, 1199-1217.

[8] Pannu, A. (2015). Artificial intelligence and its application in different areas. Artificial Intelligence, 4(10), 79-84.

[9] Muthuram, N., & Frank, F. C. (2021). Optimization of machining parameters using artificial Intelligence techniques. Materials Today: Proceedings, 46, 8097-8102.

[10] Abreu de Souza, F., Crispim Romão, M., Castro, N. F., Nikjoo, M., & Porod, W. (2023). Exploring parameter spaces with artificial intelligence and machine learning black-box optimization algorithms. Physical Review D, 107(3), 035004.

[11] Czako, Z., Sebestyen, G., & Hangan, A. (2021). AutomaticAI–A hybrid approach for automatic artificial intelligence algorithm selection and hyperparameter tuning. Expert Systems with Applications, 182, 115225.

[12] Ashena, R., Rabiei, M., Rasouli, V., Mohammadi, A. H., & Mishani, S. (2021). Drilling parameters optimization using an innovative artificial intelligence model. Journal of Energy Resources Technology, 143(5), 052110.

[13] R. Pavithra Guru & V. Vaithianathan. (2025). Grey wolf optimization (GWO) based efficient partitioning algorithm VLSI circuits for reducing the interconnections. Analog Integrated Circuits and Signal Processing,123(3),45-45.

[14] Yan Li,Miao Qian,Daojing Dai,Weitao Wu,Le Liu,Haonan Zhou & Zhong Xiang. (2024). Flow Control of Flow Boiling Experimental System by Whale Optimization Algorithm (WOA) Improved Single Neuron PID. Actuators,14(1),5-5.

[15] Liang Jintao,Wu Liang,Wang Liru,Li Guiyin,Zhang Tie,Zhang Yonghuai & Zhang Jiayang. (2024). Determination of difluoromethane (R32) by non-dispersive infrared (NDIR) spectroscopy with temperature compensation by an improved whale optimization algorithm (IWOA). Instrumentation Science & Technology,52(4),456-473.

[16] Yirong Guo,Xuefeng Zhang & Shaoqing Zhang. (2024). Research on logistics cooperative scheduling method based on multi-intelligent body system in supply chain management. Applied Mathematics and Nonlinear Sciences,9(1),