Improved Grey Wolf Optimization Algorithm Based on Arctangent Inertia Weight

Zhiqian Yun
School of Mathematical Sciences, Peking University, Beijing, China.
13462210908@163.com

Abstract. Grey Wolf Optimization (GWO) has several advantages in tackling optimization issues, but it still has some drawbacks such as slow convergence rate, low accuracy, and lack of stability. To address above drawbacks, this paper proposed an arctangent inertia weight strategy based on the arctangent function, and used the strategy to improve the GWO algorithm, thus proposing the Improved Grey Wolf Optimization Algorithm Based on Arctangent Inertia Weight (AGWO). Six classical test functions were selected to compare the convergence performance of AGWO with the other five classical swarm intelligence algorithms. The results show that AGWO has higher level of stability and computational accuracy, as well as faster convergence rate compared with the other five classical swarm intelligence algorithms.

Keywords: arctangent function; grey wolf optimization algorithm; convergence rate; swarm intelligence algorithm; inertia weights.

1. Introduction

In recent years, the rapid advancement of modern society has led to increased recognition of the swarm intelligence algorithm. This has resulted in heightened interest in the algorithm, leading to extensive research and its application in various industrial contexts. Many novel and significant swarm intelligence algorithms have been proposed, such as Particle swarm optimization (PSO) algorithm [1], Ant lion optimization (ALO) algorithm [2], Firefly optimization algorithm (FA) [3], Fruit Fly Optimization Algorithm (FOA) [4].

In 2014, Seyedali et al. [5] introduced the renowned GWO algorithm, which greatly drew inspiration from the cooperative hunting behavior of grey wolves in their natural habitat. This algorithm has then garnered significant attention and has been the subject of extensive research. GWO has been instrumental in addressing practical application issues, and numerous scholars have conducted a range of optimizations based on this fact. Huang et al. [6] improved the GWO algorithm using K-Means clustering, and applied it to help wall-climbing robots for bridge path detection. Li et al. [7] utilized the GWO algorithm to optimize the random forest prediction model, and used the optimized prediction model to solve the condenser vacuum prediction problem. Zhu et al. [8] combined GWO algorithm and adaptive scheduling strategy to enhance the stability of the scheduling system. Hu et al. [9] improved the computational speed and optimization accuracy of GWO using differential evolution algorithm and adaptive parameter adjustment strategy, and applied it to predict the security posture of the industrial Internet. Jiang et al. [10] combined GWO and support vector machines (SVM) to design water quality monitoring and rating system. Zheng et al. [11] improved the GWO algorithm by combining strategies such as Levy flight and used it to accurately predict the remaining life of lithium batteries. Wang[12] improved GWO algorithm to efficiently solve the multi-objective siting problem under natural disasters and to better balance satisfaction and economic costs. Zhang et al. [13] combined hybrid GWO algorithm and arithmetic optimization algorithm to efficiently generate class integration test sequences. In order to further solve the problems of the GWO algorithm in terms of the speed and accuracy of the optimization search, a simpler and more effective improvement method, namely AGWO, was proposed in this paper.

2. Grey Wolf Optimization Algorithm

The steps of the GWO algorithm are as follows:

(1) Initialize the relevant parameters of the algorithm. These include the maximum number of iterations N, the population size S, the spatial dimension of the population dim, as well as a, A, C, and the selected test function;

(2) Evaluate the fitness of each individual in the initial population according to the test function, and retain the positions of the top three individuals, named as α, β, γ;

(3) Update the individual positions within the population by Equations 1 - 3:

$$D_\alpha = |C_1 \cdot X_\alpha - X|,\ D_\beta = |C_2 \cdot X_\beta - X|,\ D_\delta = |C_3 \cdot X_\delta - X| \tag{1}$$

$$X_1 = X_\alpha - X, X_2 = X_\beta - X, X_3 = X_\delta - X \tag{2}$$

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

(4) Update the parameters a, A, C by Equations 4 - 6;

$$A = 2a \cdot r_1 - a \tag{4}$$

$$C = 2r_2 \tag{5}$$

$$a = 2 - 2r/N \tag{6}$$

(5) Calculate the fitness values of all individuals in the population andα, β, γ the location of the wolves to be updated;

(6) Determine whether the algorithm reaches the termination condition, if yes, output the result, otherwise go back to step (3).

3. Improved Grey Wolf Optimization Algorithm Based on Arctangent Inertia Weight

3.1 Arctangent Inertia Weight

The update formula for the inertia weights in this paper is shown in Equation 7:

$$w = w_{start} - (w_{start} - w_{end}) \cdot (arctan(b \cdot (e^{\frac{t}{50}} - c))/\pi + 1/2) \tag{7}$$

Specifically, this paper takes $w_{start} = 0.9$, $w_{end} = 0.4$, $b = 0.8, c = 25$.

When w_{start} $=w_{end} = 1,\ w = 1$.

To visualize the curve variation of the inertia weight equation of Equation 7, the graph of the inertia weight of Equation 7 is shown in Figure 1.

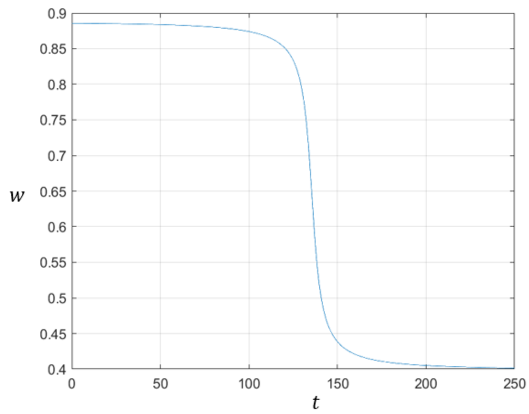

Figure 1. Arctangent Inertia Weight diagram

When the value of the number of iterations t is changed, the curve also undergoes a corresponding change. w_{end} determines the final inertia weight, which can be moderately reduced when a high speed of convergence is required; b value reflects the steepness of the change in this curve, while the change of c value is the time to change the inertia weight mutation. The larger the c value the later the algorithm will enter the local space for the search for the optimal, and thus search more extensively for the global optimum. When the parameter of w is adjusted, it is even possible to make w constant to 1.

As shown in Figure 1, at the beginning of the iteration, the inertia weight of the arctangent is the largest, which is helpful to expand the search range of the algorithm and make the algorithm not easily fall into the local optimal solution. As the number of iterations increases, the curve gradually decreases and finally stabilizes at a smaller value (i.e.w_{end}), which is helpful for the algorithm to obtain a smaller inertia weight in the later iterations and help the algorithm to find the global optimal solution in the local space in the later time. To sum up, the flexible adjustment of the inertia weight parameter of the arctangent can achieve a balance between global search and local search.

3.2 Improved Grey Wolf Optimization Algorithm Based on Arctangent Inertia Weight

The steps of AGWO are as follows:

(1) Give the initial parameters of the algorithm, which include the maximum number of iterations N, the population size S, the spatial dimension of the population dim, as well as $a,\ A,\ C$, and the inertia weights w;

(2) Evaluate the degree of adaptation of each individual in the initial population according to the test function, and retain the positions of the top three individuals with the best adaptation among them, named as $\alpha,\ \beta,\ \gamma$;

(3) Update w according to equation 7;

(4) Update of population individual positions by the sequence of Equations 1, 8 and 3;

$$X_1 = w \cdot X_\alpha - X,\ X_2 = w \cdot X_\beta - X,\ X_3 = w \cdot X_\delta - X \tag{8}$$

(5) Update parameters $a,\ A,\ C$ according to Equations 4 - 6;

(6) Recalculate the fitness values of individuals in the population and update the positions of α, $\beta,\ \gamma$;

(7) Determine whether the algorithm has reached the termination condition, if yes, output the result, otherwise repeat step (3).

The flow chart of the algorithm of AGWO is shown in Figure 2:

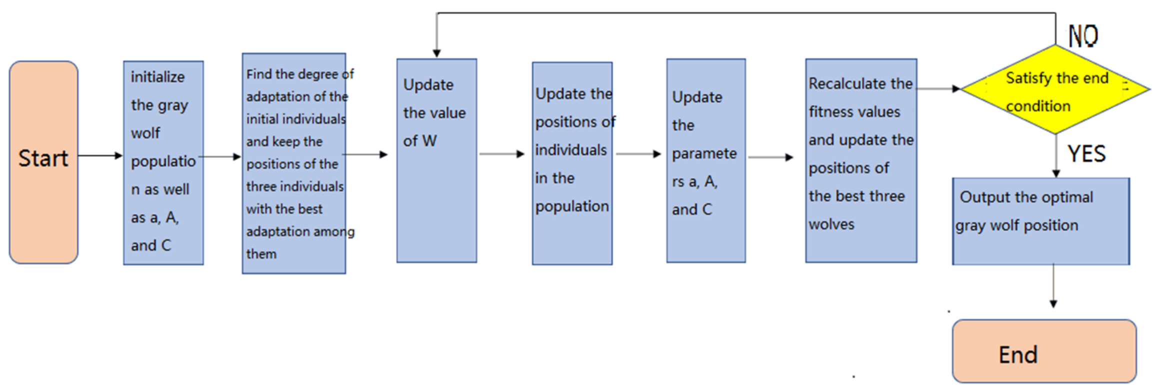

Figure 2. Flow chart of AGWO

In the algorithm of AGWO, the GWO algorithm is improved in this paper using the inertia weight of the arctangent. As the number of iterations increases, the inertia weight gradually decreases monotonically from a larger value that is more fixed at the beginning and finally stays at a smaller value. This optimization technique involves utilizing larger inertia weights during the early stages of the algorithm to enable a wider global search range for the optimal solution. As the weights gradually decrease, the search range is reduced, leading to a transition from global to local optimal search. During the later stages of the algorithm, smaller inertia weights are employed to facilitate a deeper search for the respective local optimal solution within multiple smaller ranges, ultimately resulting in a superior global optimal search outcome. Consequently, this leads to an improved search for the global optimum.

The relationship between AGWO and GWO is shown in Figure 3:

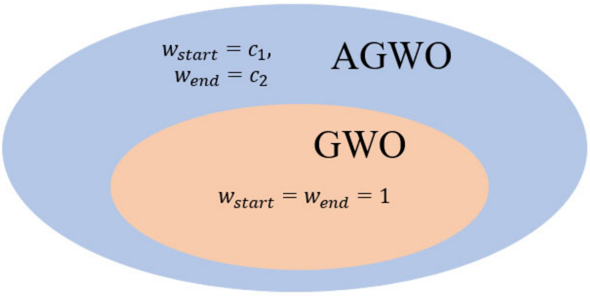

Figure 3. Relationship between AGWO and GWO

From Figure 3, it is clear that AGWO is an extension and upgrade of GWO. AGWO degenerates to GWO when wstart = wend = 1. In other words, AGWO degenerates to GWO when the inertia weight function w of the arctangent is constant to 1.

4. Simulation Experiments and Results

4.1 Experimental Design

Simulation environment: Windows 10, memory 64G, CPU Clock Speed 2.9GHz, MATLAB R2019a.

In this paper, six different test functions were selected to compare the outcome-seeking performance of AGWO with five other swarm intelligence optimization algorithms, which are shown in Table 1:

Table 1. Test function parameter design

Function Number	Function Formulas	Dimension	Variable value range	Optimum value
F1	$f(x) = \frac{1}{4000}\sum_{i-1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x}{\sqrt{i}}\right) + 1$	50,150. 450	[-600,600]	0
F2	$f(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	50,150. 450	[-5.12,5.12]	0
F3	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} + 20 + e$	50,150. 450	[-32,32]	0
F4	$f(x) = \sum_{i=1}^{n} \lvert x_i\rvert + \prod_{i=1}^{n} \lvert x_i\rvert$	50,150. 450	[-10,10]	0
F5	$f(x) = \sum_{i=1}^{n} x_i^2$	50,150. 450	[-100,100]	0
F6	$f(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i-1} x_j)^2$	50,150. 450	[-100,100]	0

In order to assess the efficacy of the AGWO algorithm as presented in this study, we opted to utilize six distinct test functions outlined in Table 1. These test functions were used to conduct a comparative analysis of six swarm intelligence algorithms, including the AGWO algorithm. The present study displayed the logarithmic mean of optimal fitness (LMOF) for a population size of 50, a maximum number of iterations of 200, and 20 separate independent runs, as represented by equation (9):

$$LMOF(t) = \frac{1}{G}\sum_{i=1}^{G} log_{10}(gbest_i(t)) \tag{9}$$

where G represents the total number of independent runs and $gbest_i(t)$ represents the optimal adaptation value for the i-th run of the t-th iteration. The present study employs logarithmic averaging as opposed to direct averaging to provide a more intuitive demonstration of the algorithm's convergence speed and accuracy towards the optimal value.

4.2 Analysis of experimental results

4.2.1 Comparison of the convergence curves of the six algorithms

The comparative convergence curves of the six algorithms based on three solution dimensions (D = 50, 150, 450) with six test functions are shown in Figures. 4-9:

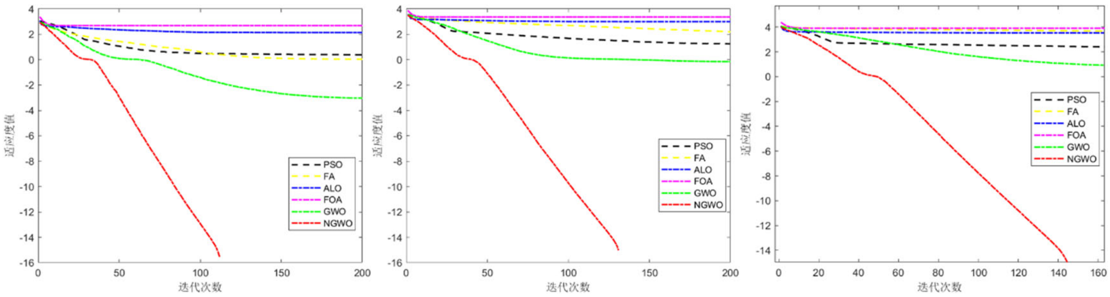

Figure 4. Comparison of average convergence curves for F1 (D=50,150,450)

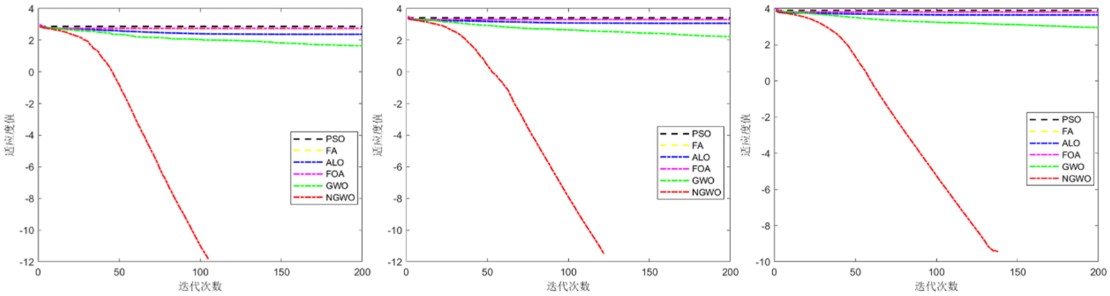

Figure 5. Comparison of the average convergence curves for F2 (D=50,150,450)

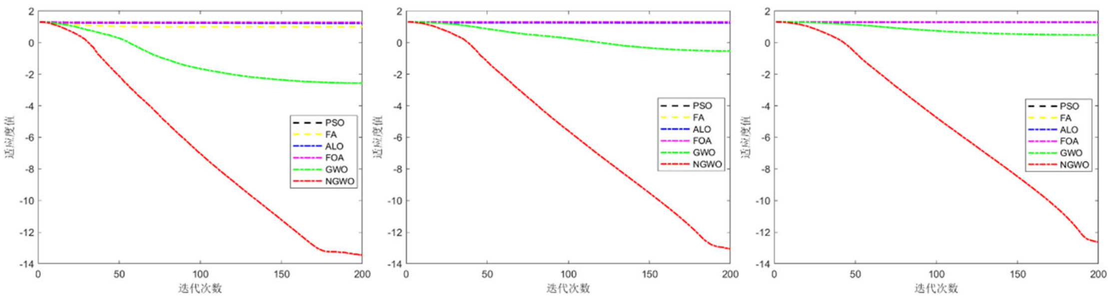

Figure 6. Comparison of the average convergence curves for F3 (D=50,150,450)

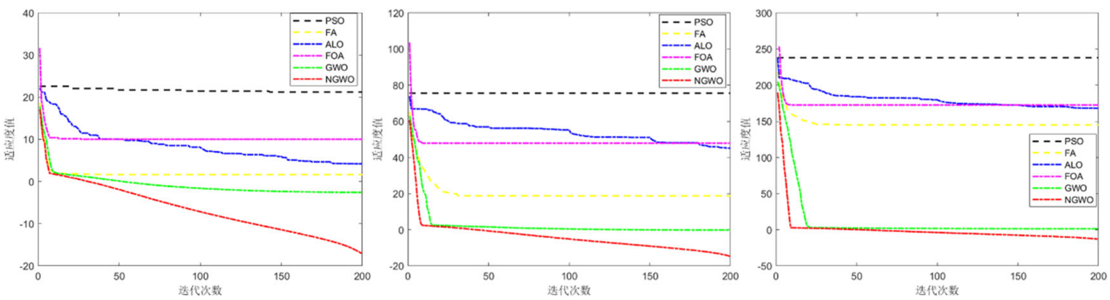

Figure 7. Comparison of the average convergence curves for F4 (D=50,150,450)

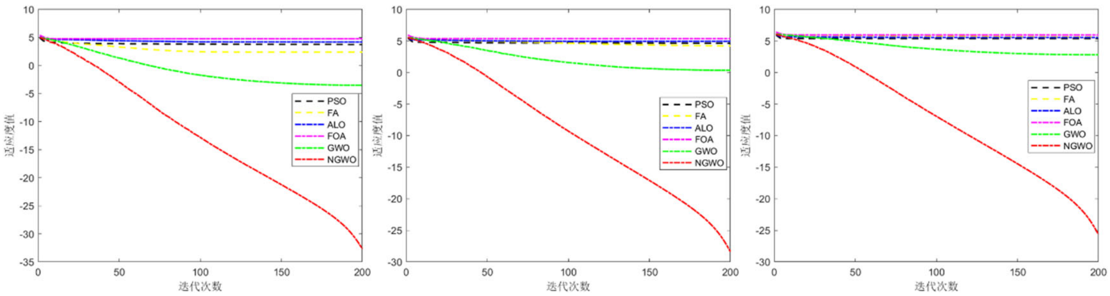

Figure 8. Comparison of the average convergence curves for F5 (D=50,150,450)

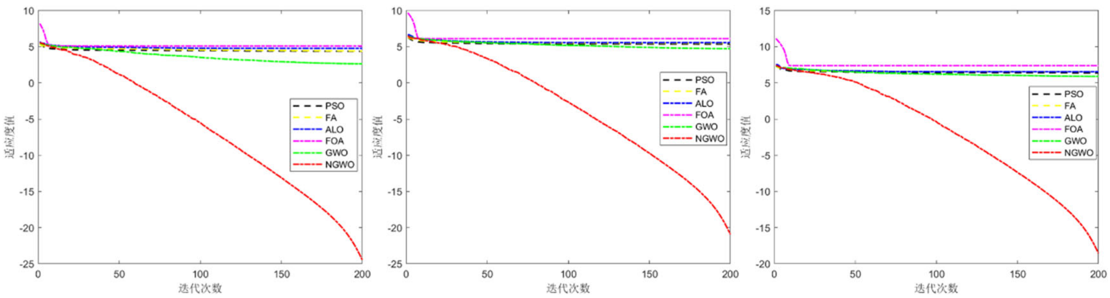

Figure 9. Comparison of the average convergence curves for F6 (D=50,150,450)

In Figures. 4-9, the horizontal coordinates represent the number of iterations t and the vertical coordinates represent the LMOF values. Based on the observations obtained from Figures 5 and 9, it can be inferred that the algorithms, with the exception of AGWO, exhibit a slow convergence rate and tend to converge towards a local optimum in both low-dimensional (D=50) and high-dimensional (D=150,450) environments. In contrast, the AGWO algorithm demonstrates a high convergence rate while approaching the optimal value, thereby possessing both accuracy and speed advantages over the other five algorithms. In Figure 4, Figure 6 and Figure 8, the results demonstrate that the AGWO algorithm exhibits favorable convergence rates and accuracy in identifying optimal values even in low-dimensional scenarios (D=50). The GWO algorithm has demonstrated some advancements, however, as depicted in Fig. 8, its curves plateau as the number of iterations progressively increases, suggesting that it gradually converges towards a local optimum. Upon analysis of the high dimensionality (D=150,450), it is evident that the convergence rate of the GWO algorithm decreases with an increase in dimensionality. Additionally, the accuracy of the solution deteriorates gradually, indicating a tendency to fall into local optima. Conversely, the AGWO algorithm stands out as the only one among the six algorithms that does not succumb to local optima in all cases. The primary aspect to examine pertains to Figure 7, which appears to indicate that the disparity between AGWO and GWO algorithms is diminishing with the escalation of dimensionality. Upon careful analysis of the figure, it is evident that the vertical coordinate's value ranges from -20 to 40 in low dimension (D=50). However, in high dimensions (D=150, 450), the value range expands to -20 to 120 and even -50 to 300. This phenomenon is primarily attributed to algorithms such as PSO, which tend to converge to a local optimum that is often mistaken for the global optimum in high dimensions. The results indicate that AGWO outperforms the other five group intelligence algorithms in terms of accuracy, and this gap widens as the dimensionality increases. Furthermore, AGWO's convergence accuracy remains significantly higher than that of other algorithms, and its convergence rate does not decrease significantly. Despite this, the gap between the GWO algorithm and AGWO does not appear to decrease significantly, as discussed in further detail in Section 3.2.2. Overall, the AGWO algorithm exhibits superior performance compared to the other five algorithms in relation to both convergence rate and optimization accuracy across diverse test functions and dimensions. Conversely, the remaining algorithms exhibit a proclivity towards local optima or succumb to local optima in high dimensions. In conclusion, among these six optimization-seeking algorithms, the AGWO algorithm demonstrates the strongest performance, and the advantage is quite significant.

4.2.2 Comparison of global optima of six algorithms

The solution dimensions were selected as 50, 150, and 450, and the experiment was conducted 20 times with a maximum iteration limit of 200. Table 2 presents a comparison of the mean and standard deviation of the optimal adaptation values for the six algorithms under assessment.

Table 2. Comparison of mean and standard deviation of 6 algorithms

Test Functions	Algorithm	D=50		D=150		D=450	
		Average value	Standard deviation	Average value	Standard deviation	Average value	Standard deviation
F1	PSO	2.397361	0.183597	17.71892	2.123066	215.2611	18.61186
	FA	1.089424	0.041972	170.3552	72.69863	4173.686	220.6671
	ALO	144.7912	55.92599	999.6478	108.1184	3421.179	542.1455
	FOA	496.4608	66.02716	2333.651	109.2987	8062.722	305.9886
	GWO	0.02096	0.046116	0.711684	0.10133	6.849405	0.980065
	AGWO	0	0	0	0	0	0
F2	PSO	797.0801	50.21154	2568.438	85.58295	7940.017	55.31729
	FA	473.3806	26.61415	1943.143	82.88611	6714.991	142.8697
	ALO	224.3171	47.32076	1115.679	51.8601	4400.808	179.0715
	FOA	579.099	23.45313	2010.699	57.2445	6547.157	134.8731
	GWO	38.51719	19.36952	160.38	27.02547	1044.084	62.53862
	AGWO	0	0	0	0	0	0
F3	PSO	19.09047	0.269401	20.20524	0.106444	20.66588	0.832217
	FA	10.34589	1.834963	19.2756	0.526897	20.56815	0.092304
	ALO	17.19982	0.934663	17.96652	1.637775	19.12606	0.35587
	FOA	19.46162	0.275883	20.2394	0.052002	20.4877	0.028138
	GWO	0.002826	0.000951	0.325099	0.179148	3.089382	0.145766
	AGWO	3.31E-14	8.85E-15	8.86E-14	1.29E-14	2.50E-13	6.39E-14
F4	PSO	2.12E+22	3.65E+22	2.92E+77	5.56E+77	1.13E+242	Inf
	FA	47.65927	6.750237	1.82E+22	2.98E+22	2.97E+151	6.65E+151
	ALO	4.03E+10	9E+10	2.97E+55	6.41E+55	2.82E+219	Inf
	FOA	1.17E+12	2.52E+12	5.30E+51	7.91E+51	3.31E+175	Inf
	GWO	0.003134	0.001193	0.742007	0.206584	22.99915	2.895033
	AGWO	7.99E-18	3.20E-18	2.17E-15	5.09E-16	9.02E-14	1.83E-14
F5	PSO	5471.571	659.2572	43056.22	4669.87	218625	10498.82
	FA	284.1331	154.9504	14404.29	4815.736	432997.4	29278.87
	ALO	16941.54	6116.736	84481.83	18491.12	403505.4	55513.96
	FOA	59849.31	5218.677	243903.9	7369.317	906830.3	20103.68
	GWO	0.000337	0.000163	2.966302	0.661092	683.5063	77.66916
	AGWO	2.29E-33	9.19E-34	4.46E-29	1.40E-29	2.34E-26	5.06E-27
F6	PSO	32163.69	6482.368	235265.1	29374.49	2419042	494562
	FA	17246.96	4391.978	324065.3	43070.01	3785401	1630313
	ALO	45417.43	11361.38	383858	118924.5	4023747	1967333
	FOA	152303.4	30163.54	1.32E+06	129089	25389484	3955484
	GWO	732.0458	452.2643	56192.11	9749.634	806848.7	133197.8
	AGWO	4.13E-25	6.53E-25	2.53E-21	3.45E-21	4.44E-19	4.58E-19

Table 2 presents a comparative analysis of six algorithms in terms of their ability to identify the optimal solution for six test functions in three dimensions, with varying dimensions (D=50,150,450). Table 2 demonstrates that the AGWO algorithm attains the optimal solution for F1 and F2 tests. However, for the remaining test functions, the AGWO algorithm does not achieve the optimal solution. Nevertheless, the optimal solution obtained by AGWO is relatively optimal compared to the outcomes of the other five algorithms, and further iterations may lead to the attainment of the optimal solution. This table also provides further insight into the issue illustrated in Figure 7 of the preceding section. Specifically, it demonstrates that the ratio between the logarithmic values of the mean values of the AGWO and GWO algorithms remains largely constant as the dimensionality of F4 increases. This finding serves to confirm that the discrepancy between the two algorithms depicted in the aforementioned figure is primarily attributable to the visual distortion caused by the range of values on the vertical axis. Table 2 demonstrates that AGWO exhibits significantly smaller values for both mean and standard deviation when compared to the other five classical swarm intelligence algorithms. This suggests that AGWO not only surpasses the other algorithms in terms of convergence rate and optimization-seeking accuracy, but also exhibits superior stability. The present study demonstrates that the enhanced concept presented in this paper has resulted in a discernible enhancement in the efficacy of the Grey Wolf Optimization (GWO) algorithm. To summarize, the AGWO algorithm exhibits superior performance compared to alternative algorithms in regards to convergence rate, optimization accuracy, and solution stability. Additionally, it serves as a commendable optimization of the GWO algorithm.

4.2.3 Comparison of global optima of 6 algorithms

The time complexity of the GWO algorithm is O(NMD), where N is the number of grey wolves, M is the number of iterations of the algorithm, and D is the dimension of the solution. The AGWO algorithm employs Equation 7 to enhance the inertia weighting approach of the arctangent function and Equation 8 to revise the grey wolves' positions. This results in a consistent multiplication on the positions of the three optimal grey wolves during each iteration of the GWO algorithm, as compared to the GWO algorithm. To summarize, it can be stated that the AGWO algorithm does not result in an increase in the time complexity of the GWO algorithm. Therefore, the time complexity of the AGWO algorithm remains at O(NMD). The superior performance of the AGWO algorithm in terms of merit-seeking is evident from the findings presented in Section 3.2.1 and Section 3.2.2, surpassing that of GWO. To clarify, it can be explained that the AGWO algorithm exhibits superior convergence rate, solution accuracy, and stability in comparison to the GWO algorithm. The AGWO algorithm exhibits a consistent time complexity and demonstrates superior performance compared to the GWO algorithm, when both algorithms are implemented with identical population size and solution dimension.

5. Summary and Outlook

The present study introduces the AGWO as a potential solution to address the limitations of the GWO algorithm, such as inadequate convergence rate and susceptibility to local optimality. The AGWO algorithm utilizes modified inertia weights through the implementation of the arctangent function to achieve a balance between extensive global search capabilities during the initial stage and efficient local search capabilities during the later stage. This algorithm has demonstrated favorable search efficiency. The empirical findings indicate that AGWO outperforms five other swarm intelligence algorithms in terms of convergence rate, search accuracy, and solution stability. Moreover, the enhancement is straightforward and feasible to integrate into everyday applications and code programming, thereby possessing practical application value and potential.

Simultaneously, it is imperative to acknowledge that despite the numerous benefits of the AGWO algorithm, it is not immune to certain limitations. The inertia weight function is influenced by various explanatory variables that are contingent on the environment in which it is applied. These variables include the number of iterations and other constraint-related conditions. The AGWO algorithm may encounter challenges in achieving widespread applicability due to the need for adjustments to response parameters in response to varying practical requirements. Simultaneously, it can be observed that the AGWO algorithm's inertia weights are contingent solely upon the quantity of iterations required for modifications.

Additional investigation is required to address the aforementioned issues identified by the AGWO algorithm. Specifically, the development of adaptive inertia weighting techniques that can be flexibly modified based on the population's fitness level is necessary. One possible avenue for improvement involves the adaptive adjustment of parameter values within the utilized inertia weight function, tailored to the specific environmental conditions, thereby expanding its potential applicability across a wider range of contexts.

References

[1] HOLLAND J H. Adaptation in natural and artificial systems [M]. Michigan: The University of Michigan Press, 1975.

[2] MIRJALILI S. The ant lion optimizer [J]. Advances in Engineering Software, 2015, 83: 80-98.

[3] Yang X S. Firefly algorithms for multimodal optimization [C]. International Symposium on Stochastic Algorithms. Springer, Berlin, Heidelberg, 2009: 169-178.

[4] PAN W T. A New Evolutionary Computation Approach: Fruit Fly Optimization Algorithm[C]. 2011 Conference of Digital Technology and Innovation Management, Taipei, 2011.

[5] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69: 46-61.

[6] Huang H X, Yu G W, Cheng S S, Li C M. Full coverage path planning of bridge inspection wall-climbing robot based on improved grey wolf optimization algorithm [J/OL]. Journal of Computer Applications:1-8 [2023-05-25].

[7] Li W F, Ding X, Fang J J. A GWO-RF-based method for condenser vacuum prediction[J]. Journal of Power Engineering,2023,43(04):436-442.

[8] Zhu Y H, Luo L F, Lu Q H, Qin M Y, Wang J H, Wei H L, Guo W H. Flexible job shop dynamic scheduling method for order disturbance [J/OL]. Computer Integrated Manufacturing Systems:1-23 [2023-05-25].

[9] Hu X D, Lv G F, Bai Y. A Method of Security Situation Prediction for Industrial Internet Based on Optimized Support Vector Regression[J]. Acta Electronica Sinica, 2023, 51(02):446-454.

[10] Jiang D, Xiao M H, Zhang N, Zhou J B, Zhu H, Wang S, Chen S H. Water quality monitoring and grade judgment system based on IGWOPSO-SVM algorithm [J/OL]. Journal of South China Agricultural University, 2023(04):1-18[2023-05-25].

[11] Zheng Q G, Yang X G, Liu D, Li X. Lithium battery remaining life prediction method based on improved grey Wolf optimization least squares support vector machine [J/OL]. Journal of Chongqing University:1-13 [2023-05-25].

[12] Wang F Y, Wang X R. Research on the location of emergency material center with two-stage and multiobjective under sudden natural disasters [J/OL]. Journal of Safety and Environment:1-14 [2023-05-25].

[13] Zhang W N, Zhou Q L, Jiao C YG, Xu T. Hybrid Algorithm of Grey Wolf Optimizer and Arithmetic Optimization Algorithm for Class Integration Test Order Generation[J]. Computer Science, 2023, 50(05):72-81.