Adaptive Differential Whale Optimization Algorithm

Renxing Chen[1], Yue Hu[2], Xinyuan Zhang[3], Jianming Jiang[4]
[1] *School of Intelligent Systems Science and Engineering*, *Jinan University*, Guangzhou, P. R. China, chenrenxing20@stu2020.jnu.edu.cn
[2] *College of Engineering Sciences*, *Hanyang University*, *ERICA*, South Korea, capue@hanyang.ac.kr
[3] *School of Intelligent Systems Science and Engineering*, *Jinan University*, Guangzhou, P. R. China, zhangxy@jnu.edu.cn
[4] *Zhejiang Normal University*, Jinhua, P. R. China, Jiangvic2021@163.com
(Corresponding author: Jianming Jiang, E-mail: Jiangvic2021@163.com)

***Abstract*—The whale optimization algorithm (WOA) is effective for solving complex engineering optimization problems, but it often converges slowly and easily gets trapped in local optima. To address these issues, this paper proposes an adaptive differential WOA (ADWOA) with five improvement strategies. A dynamic convergence factor is used to balance exploration and exploitation. Adaptive inertia weights enhance both global and local search. A refined search mechanism enables information exchange across dimensions to improve local accuracy. Laplace-distribution perturbation helps maintain population diversity, and a differential evolution operator is integrated to strengthen search performance. ADWOA is tested on standard benchmark functions, and the results show that it achieves better solution quality and faster convergence than existing methods.**

***Keywords*—Whale optimization algorithm, Differential Operator, Adaptive inertia weights**

I. Introduction

Optimization problems arise widely in science and engineering. As problem dimensionality, scale, and nonlinearity increase, modeling and solving such problems becomes more difficult. Traditional optimization methods, such as gradient descent [1], Newton's method [2][3], the conjugate gradient method [4], and the Lagrange multiplier method [5], offer fast and accurate convergence but depend on strong conditions, including differentiability and full knowledge of constraints. As a result, they are easily trapped in local optima and often perform poorly in complex real-world applications.

To overcome these issues, metaheuristic algorithms combine stochastic exploration with local exploitation. Swarm intelligence algorithms, inspired by collective biological behavior, include ACO [6], PSO [7], AFS [8], FA [9], CS [10], GWO [11], FOA [12], SSA [13], HHO [14], SMA [15], AO [16], AHA [17], NOA [18], and CPO [19]. Among them, WOA [20] is widely used due to its simple structure, fast convergence, and good accuracy [21]. However, like many population-based methods, WOA still suffers from premature convergence, local optima entrapment, and limited population diversity. Many studies have introduced improvements, including chaotic mechanisms [22], genetic operators [23], optimality-based feedback [24], symbiotic phases [25][28], nonlinear convergence factors [26][27], and opposition-based learning (OBL) [29][30]. WOA has also been combined with genetic algorithms [31], seagull attack models [32], and sea-squirt strategies [33] to enhance global search ability and robustness. WOA and its variants have been applied in many domains. In electrical engineering, CWOA improves solar cell parameter estimation [34], and hybrid WOA methods address multi-objective power scheduling [35]. In feature selection and clustering, chaotic WOA enhances brain tumor image classification [36], and improved WOA strategies support network community detection [37]. In image processing, WOA-based multilevel thresholding achieves high segmentation accuracy [38][39]. In path planning, WOA variants enhance obstacle avoidance for mobile robots [40] and underwater vehicle navigation in uncertain environments [41].

To address the inherent limitations of WOA, this paper proposes an adaptive differential whale optimization algorithm (ADWOA) that integrates five enhancement strategies: a dynamic convergence factor, adaptive inertia weights, an improved search mechanism, Laplace-distribution perturbation, and a differential evolution (DE) operator. The dynamic convergence factor improves the balance between exploration and exploitation across different search stages. Adaptive inertia weights strengthen global and local search by using population ranking and distance information. The refined search mechanism supports more accurate local exploration by promoting information exchange across solution dimensions. Laplace-distribution perturbation maintains population diversity and reduces premature convergence by adding controlled random variations. The DE operator further improves solution quality by using information from high-performing individuals, thus enhancing overall search efficiency. ADWOA is evaluated on benchmark functions against four classical metaheuristics and five recent WOA variants. Ablation studies, stability tests, and Wilcoxon rank-sum analyses further confirm its effectiveness.

The rest of this paper is organized as follows. Section II reviews the WOA. Section III introduces the proposed ADWOA. Section IV presents the experimental results. Section V describes engineering applications, and Section VI concludes the paper and outlines future work.

II. Classical Whale Optimization Algorithm

In WOA, the position of each whale represents a solution vector, and the algorithm simulates this behavior through three key strategies: encircling the prey, executing a bubble net attack, and searching for prey.

A. Encircling Prey

Humpback whales possess the capability to detect and encircle prey locations. Similarly, in WOA, as the precise location of the optimal solution within the search space is typically unknown, the target prey is presumed to represent the best candidate solution or one very close to the optimal solution.

Consequently, once the optimal search agent has been identified and defined, subsequent phases entail other search agents endeavoring to adjust their positions towards that optimal agent. This behavioral adaptation is mathematically represented by (1) and (2) [20], where **X** represents the position vector, t denotes the number of current iterations, $\mathbf{X}_{best}$ denotes the position vector of the best solution, and an update of the best solution will be performed after each iteration. Parameters *maxIter* is the maximum number of iterations, and r represents a random value between 0 and 1.

$$\mathbf{D} = C \cdot \mathbf{X}_{best}(t) - \mathbf{X}(t) \tag{1}$$

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

$$A = a \cdot (2 \cdot r - 1) \tag{3}$$

$$C = 2 \cdot r \tag{4}$$

$$a = 2 \cdot (1 - t / maxIter) \tag{5}$$

B. Spiral Bubble-net Attacks

Two strategies, the shrink-wrap mechanism and the spiral renewal mechanism, were specifically described to mathematically model the bubble net feeding behavior of humpback whales. The first one is shrinking encirclement mechanism. When the coefficient vector A is less than or equal to 1, the individual whale gradually approaches the current best whale, circling the prey in smaller steps while seeking a better position. Conversely, when A is greater than 1, the search agent has the flexibility to randomly explore regions beyond the current optimum, thus demonstrating the algorithm's ability to perform global search. The shrink-wrap mechanism extends the optimal solution region, balancing between search and exploitation to facilitate efficient optimization. The second one is spiral renewal mechanism. In this phase, the whale first evaluates the distance between itself and the best position obtained so far, and then moves upstream in a spiral fashion while spitting out bubbles of different sizes to trap fish and shrimp. This behavior is represented by (6).

$$\mathbf{X}(t+1) = \mathbf{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \mathbf{X}_{best}(t) \tag{6}$$

Where $\mathbf{D}'$ denotes the distance of an individual $\mathbf{D}(t)$ from the whale in the optimal position up to the present time, and b is a constant defining the shape of the logarithmic spiral. Parameter l is a random number in the interval [-1, 1]. When a humpback whale attacks a prey with a bubble net, the humpback whale swims along a spiral path around the prey at the same time, i.e., both strategies are carried out simultaneously. To obtain this model, it is assumed that there is a 50% probability of choosing either the shrink-wrap mechanism or the spiral update mechanism to update the position of the whale during the optimization process. The mathematical model is shown in (7), where p is a random number within the interval [0, 1].

$$\mathbf{X}(t+1) = \begin{cases} \mathbf{X}_{best}(t) - A \cdot \mathbf{D} & if\ p < 0.5 \\ \mathbf{D}' \cdot e^{bl} \cdot cos(2\pi l) + \mathbf{X}_{best}(t) & otherwise \end{cases} \tag{7}$$

C. Search for Prey

In WOA, an exploitation process prevents the algorithm from falling into a local optimum. In the search for prey mechanism, randomly selected individuals guide the evolution of individuals performing the search task in the exploration phase, enriching the diversity of the population. The mathematical representation of this phase is shown in (8) and (9), where $\mathbf{X}_{rand}$ denotes the position vector of a randomly selected individual whale from the current population, and $\mathbf{D}_{rand}$ denotes the distance from the randomly selected individual whale to its prey.

$$\mathbf{D}_{rand} = | C \cdot \mathbf{X}_{rand}(t) - \mathbf{X}(t) | \tag{8}$$

$$\mathbf{X}(t+1) = \mathbf{X}_{rand}(t) - A \cdot \mathbf{D}_{rand} \tag{9}$$

III. Differential Whale Optimization Algorithm

A. Dynamic Convergence Factor

In the classical WOA, the coefficient parameter value of A is utilized to balance the global and local search capabilities of the algorithm. When $A > 1$ the algorithm enhances the global search capability by expanding the search scope to find a better candidate solution. Conversely, when $A \leq 1$ the algorithm narrows down the search scope and performs a fine search in the local region. From (2), the value of A is constrained by a. When a is large, the algorithm performs a global search and has a better ability to escape from the local optimum. Conversely, the algorithm exhibits stronger local search ability and faster convergence when a is small. In WOA, the convergence factor linearly decreases with the increase of the number of iterations from 2 to 0. However, the new strategy of linearly changing the convergence factor a does not fully reflect the optimization search process of the actual WOA. Therefore, this paper proposes a nonlinear convergence factor to prevent late convergence into local optima and balance the algorithm's global and local search ability. This paper proposes a mathematical expression based on the sigmoid function's nonlinear trend in (10), where m and n are optional parameters, and t is the current number of iterations.

$$a(t) = \frac{2}{1 + t / maxIter \cdot m \cdot e^{n}} \tag{10}$$

B. Adaptive Inertia Weights

The manipulation of inertia weight coefficients within the Whale Optimization Algorithm (WOA) serves as a pivotal mechanism for refining the balance between exploration and exploitation capabilities. In practice, larger inertia weight coefficients correspond to augmented search steps, thus bolstering the algorithm's capacity to navigate away from local optima and engage in broader global exploration endeavors. Conversely, smaller inertia weight coefficients facilitate finer-scale local searches, thereby enhancing algorithmic precision and expediting convergence rates. Nonetheless, a noteworthy limitation of the conventional WOA lies in the static nature of inertia weights, which remain fixed at a value of 1 throughout the iterative process. While this approach effectively supports global exploration, it inadvertently constrains the algorithm's ability to conduct nuanced searches in proximity to the global optimum.

To address this constraint, this study leverages insights from the PSO algorithm, introducing an adaptive weight parameter that dynamically adjusts based on the current iteration count and individual rank within the population. The mathematical expressions governing this adaptive weight system are detailed in (11) and (12), wherein dynamic factor θ, maximum threshold value θ_{max}, minimum threshold value θ_{min}, individual fitness value ranking in the population $rank_i$, and population size NP play distinctive roles. By integrating these adaptive inertia weights with the incumbent WOA (2) and (9), the whale positions undergo updates in accordance with (13) and (14). The introduction of adaptive inertia weights engenders a nuanced search dynamics within the algorithmic process. Initially, the larger early search step size rapidly diminishes, thereby amplifying global exploration capabilities and hastening the convergence pace. Subsequently, the adaptive weight gradually decreases as the algorithm progresses, prompting individuals to converge toward a focal point akin to a pilot whale. This shift facilitates effective local search exploration by leveraging the diminished adaptive weight, ergo enhancing local search capabilities. The ranking of individuals within the population inherently reflects their relative superiority or inferiority, conferring lower-ranked individuals with enhanced global exploration potential, while positioning higher-ranked individuals as apt candidates for local search roles. By recalibrating the dynamic factor minima based on individual rankings, equitable representation and influence across the population spectrum are ensured, thereby safeguarding diversity and circumventing premature entrapment within local optima.

$$\theta_{min} = \frac{rank_i}{NP} \cdot \theta_{max} \tag{11}$$

$$w(t) = \theta_{max} - \frac{t}{MaxIter} \cdot (\theta_{max} - \theta_{min}) \tag{12}$$

$$\mathbf{X}(t+1) = w(t) \cdot \mathbf{X}_{best}(t) - A \cdot \mathbf{D} \tag{13}$$

$$\mathbf{X}(t+1) = w(t) \cdot \mathbf{X}_{rand}(t) - A \cdot \mathbf{D}_{rand} \tag{14}$$

C. Differential Search

The conceptual framework of the differential search operation put forth in this study draws upon the foundational tenets of the differential evolution algorithm [42]. At its core, this operation harnesses the current optimal individual within the whale population to engender a novel individual through the stochastic differencing process involving two randomly selected individuals from the same cohort. The resulting individual bears the hallmark of enhanced diversity, a pivotal attribute that shields the algorithm from being ensnared in suboptimal local optima. Noteworthy is the pivotal role of the mutation operation, a key point of departure distinguishing the differential evolution algorithm from other evolutionary algorithms in the realm of mutated individual generation. Central to our study is the strategic use of optimal individuals to catalyze the creation of mutated entities during the advanced stages of algorithmic execution, as delineated in (10). Here, $\mathbf{X}_{\text{best}}$ designates the apex position within the extant whale population, while p and q represent two stochastic variables drawn from the current whale cohort. Characterizing this process is the predefined scaling factor F.

$$\mathbf{V}_i = \mathbf{X}_{best} + F \cdot (\mathbf{X}_p - \mathbf{X}_q) \tag{15}$$

After the mutation phase, a pivotal crossover operation ensues, wherein the mutants engendered in the prior stage interface with the original individuals, giving rise to a hybridized population that interlaces mutant elements with specified probability, as detailed in (16). The crossover factor C_r, modulated within the inclusive range of [0,1], exerts a critical influence in shaping the genetic exchange dynamics. As the algorithmic sequence unfolds, a judicious selection process comes to the fore, predicated upon a comparative assessment of fitness metrics between the emergent entity $\mathbf{Z}$ and the extant individuals, guided by the tenets of a nuanced greedy principle. The individual exhibiting superior fitness emerges triumphant, propelling it forward to commence the subsequent iteration cycle.

$$\mathbf{Z}_i = \begin{cases} \mathbf{V}_i & if\, rand(0,1) < Cr \\ \mathbf{X}_i & otherwise \end{cases} \tag{16}$$

D. Laplace Distribution Perturbation Strategy

As iterations progress in WOA, population diversity decreases, increasing the risk of premature convergence. To address this issue, Laplace-distribution perturbation is introduced to add controlled randomness and help individuals escape local optima. This mechanism broadens the search space, improves diversity, and strengthens global search capability. The Laplace distribution is symmetric around its mean and has a sharp peak with heavy tails. Its probability density function, shown in (17), is defined by the location parameter μ and the scale parameter λ. A larger λ produces a flatter and more dispersed distribution, while a smaller λ yields a sharper curve with more concentrated probability. In this study, μ is set to 0 so that the peak is centered at the origin. Applying Laplace perturbation to individuals in later iterations helps the algorithm move away from local optima by sampling nearby regions and increasing population diversity. Extremely large λ values may cause excessive variation and reduce optimization stability, while very small λ values limit diversity. Thus, λ is set to 1 to maintain an appropriate balance. The perturbation strategy is defined in (18), where random numbers drawn from Laplace (0,1) are used to adjust individual positions. This approach enhances diversity and effectively reduces premature convergence.

$$f(x) = \frac{1}{2\lambda} \cdot e^{-\frac{|x-\mu|}{\lambda}} \tag{17}$$

$$\mathbf{X}_i(t) = \mathbf{X}_i(t) \cdot \left(1 + Laplace(0,1)\right) \tag{18}$$

E. Purifying Search Strategy

In this approach, the population undergoes a bifurcation based on the hierarchical ranking of individual fitness levels. The top five individuals form the first segment, serving as the elite cohort, while the remaining individuals constitute the second segment. Drawing inspiration from the concept of purified search as delineated in prior literature [43], individuals within the elite segment engage in a specialized exchange of insights across three distinct dimensions. This mechanism is designed to expedite the convergence towards the global optimum by intensifying exploration of potential optimal solution regions, thereby accelerating the overall convergence velocity of the algorithm.

To systematically evaluate the impact of different dimensions on an individual's optimal fitness, a measure of relative importance is established by comparing the effects of any two dimensions within an individual. Specifically, for dimensions p and q, randomly selected from an individual $\mathbf{X}_i$ such that $\mathbf{X}_i^p \neq \mathbf{X}_i^q$, the designation $\mathbf{X}_i^{\sigma(p)=q}$ represents a new individual generated by substituting the value in the pth dimension of $\mathbf{X}_i$ with the value in the qth dimension. If the fitness value of $\mathbf{X}_i^{\sigma(p)=q}$ surpasses that of $\mathbf{X}_i$, it is inferred that the values in the pth dimension hold greater significance than those in the qth dimension. The process of purification search harnesses these comparative assessments to generate new individuals with prospective advantages, followed by the evaluation of new and original individuals based on a greedy selection approach outlined in (19) for progression into the subsequent iteration.

$$\mathbf{X}_i(t+1) = \begin{cases} \mathbf{X}_i & \text{if } f(\mathbf{X}_i) < f(\mathbf{X}_i^{\sigma(p)=q}) \\ \mathbf{X}_i^{\sigma(p)=q} & \text{otherwise} \end{cases} \tag{19}$$

This intricate interplay of differential search, mutation, crossover, and selection operations undergirds the algorithmic prowess, orchestrating a seamless trajectory towards convergence within the optimization landscape. Utilizing the five strategies outlined in the preceding subsections, the flowchart for the ADWOA methodology is depicted in Fig. 1, providing a concise yet comprehensive representation of the algorithmic framework.

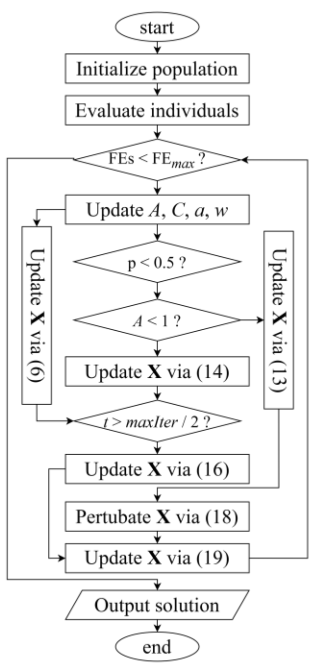

Fig. 1. Flowchart of ADWOA

IV. Experimental Results

In this section, we tested 23 benchmark functions [44] to validate the effectiveness of ADWOA. All compared algorithms are conducted in 30 independent trials and the results are averaged over the trials. The average and standard deviation values of the results are represented by avg., and std., respectively. Two-sided Wilcoxon rank sum test is conducted at significance level α = 0.05, the highlighted entries are significantly better. Furthermore, this section presents a variety of comparative experiments, assessing the performance of different algorithms through convergence curves and measuring minimum, mean, and standard deviation values. The simulation experiments were conducted on a computer with an Intel(R) Core(TM) i5-13500HX@2.5 GHz CPU, running Windows 11.

A. Comparison with Other Variants Of WOA

The experimental results are summarized in Table I. Overall, the data show that ADWOA performs strongly compared with contemporary algorithms, including Dynamic Gauss Salp Swarm Optimization (DGSWOA) [45], Improved WOA (ImWOA) [46], WOA with Sinh Rectified Linear Function (WOASCALF) [47], Multi-Swarm Fireworks WOA (MSFWOA) [48], and the Salp Swarm Optimization Algorithm (SWOA) [49], across a wide set of benchmark functions.

For single-peaked functions, the proposed algorithm achieves the best results on all functions except F_5. For F_5, ADWOA performs slightly lower than WOASCALF and is close to SWOA. However, ADWOA still surpasses SWOA in both mean and variance. These results indicate that ADWOA is effective at exploring the solution space of unimodal functions and can reliably locate optimal values in high-dimensional settings while maintaining stable performance. For high-dimensional multi-peaked functions, the challenge increases because the number of local optima grows with dimensionality. These functions contain many local optima surrounding a single global optimum, making it harder for algorithms to avoid local traps and reach the true global solution. The experimental results pertaining to the assessment of fixed-dimensional multi-peaked functions are meticulously delineated in Table I, portraying ADWOA as the frontrunner in test functions F_{15}, F_{16}, F_{17}, F_{18}, F_{21}, F_{22}, and F_{23} by garnering the highest rank. However, within the purview of test function F_{14}, while commensurate with other algorithms in attaining the minimum value, this study's algorithm finds itself in a relatively inferior position owing to subpar mean and standard deviation outcomes. Furthermore, the evaluation of test functions F_{19} and F_{20} reveals that ADWOA exhibits marginally inferior results vs its algorithmic counterparts. This nuanced dynamic can be attributed to the intrinsic attributes of the test functions characterized by fixed dimensions and a restricted range of variables. Noteworthy is the pronounced competitive edge exhibited by ADWOA in the realm of high-dimensional functions; nevertheless, the simplicity underpinning the structures of functions in Table I fosters a scenario where most algorithms deliver satisfactory outcomes, thereby attenuating the salience of ADWOA's advantages.

B. Ablation Study

This section undertakes an empirical validation to ascertain the efficacy and superiority of the five enhanced strategies, delineating an in-depth analysis of each strategy's individual contribution towards the algorithmic performance. The WOA algorithm, augmented with the dynamic convergence factor

TABLE I
EXPERIMENTAL ON 23 CLASSICAL BENCHMARK FUNCTIONS WITH DIFFERENT WOA VARIANTS.

		WOA	DGSWOA	ImWOA	WOASCALF	MSFWOA	SWOA	DWOA
F1	avg	1.28E-20	**0.00E00**	1.09E-56	2.48E-189	2.62E-59	2.04E-37	**0.00E00**
	std	6.61E-20	0.00E00	5.13E-56	0.00E00	8.99E-75	3.99E-37	0.00E00
	p-value	1.86E-09	/	1.86E-09	1.86E-09	1.86E-09	1.86E-09	/
F2	avg	6.56E-15	3.05E-206	9.21E-40	5.70E-91	1.19E-28	3.88E-22	**0.00E00**
	std	1.02E-14	0.00E00	4.49E-39	3.12E-90	2.28E-44	4.67E-22	0.00E00
	p-value	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	/
F3	avg	2.50E-02	**0.00E00**	6.88E+03	4.34E-169	1.47E-59	2.00E+03	**0.00E00**
	std	1.07E-01	0.00E00	2.37E+03	0.00E00	2.28E-44	2.39E+03	0.00E00
	p-value	1.86E-09	/	1.86E-09	1.86E-09	1.86E-09	1.86E-09	/
F4	avg	2.50E-04	1.56E-192	6.98E-06	1.02E-95	5.98E-31	5.06E-02	**0.00E00**
	std	5.68E-04	0.00E00	8.85E-06	4.04E-95	1.78E-46	4.23E-02	0.00E00
	p-value	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	/
F5	avg	2.76E+01	2.83E+01	2.83E+01	**2.85E-02**	2.87E+01	1.53E+02	2.50E+01
	std	8.68E-01	3.13E-01	3.10E-01	5.63E-02	0.00E00	5.53E+02	1.86E-01
	p-value	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	6.26E-01	/
F6	avg	1.55E00	3.38E00	6.19E-01	3.09E-04	2.37E00	1.40E-07	**2.81E-10**
	std	6.38E-01	3.96E-01	2.30E-01	7.19E-04	0.00E00	1.02E-07	4.00E-10
	p-value	1.86E-09	1.86E-09	1.86E-09	3.73E-09	1.86E-09	1.86E-09	/
F7	avg	3.71E-04	1.31E-04	3.05E-04	1.25E-04	5.75E-02	7.61E-03	**8.46E-05**
	std	3.77E-04	2.06E-04	3.08E-04	1.04E-04	1.41E-17	3.15E-03	8.91E-05
	p-value	1.86E-09	3.60E-01	1.86E-09	1.00E-01	1.86E-09	1.86E-09	/
F8	avg	-6.98E+03	-6.87E+03	-1.25690142E+04	-1.25694806E+04	-4.46E+03	-8.19E+03	**-1.25694841E+04**
	std	3.22E+02	1.68E+02	8.95E-01	9.57E-03	9.25E-13	5.18E+02	7.25E-03
	p-value	1.86E-09	1.86E-09	7.01E-05	4.49E-02	1.86E-09	1.86E-09	/
F9	avg	1.70E-14	**0.00E00**	1.32E-14	**0.00E00**	**0.00E00**	2.27E+01	**0.00E00**
	std	3.04E-14	0.00E00	3.23E-14	0.00E00	0.00E00	5.57E00	0.00E00
	p-value	6.66E-03	/	3.84E-02	/	/	1.86E-09	/
F10	avg	1.84E-11	4.44E-16	6.12E-15	**4.44E-16**	3.99E-15	1.90E-03	**4.44E-16**
	std	4.71E-11	1.00E-31	3.56E-15	1.00E-31	8.02E-31	1.03E-02	1.00E-31
	p-value	1.86E-09	/	2.27E-06	/	1.86E-09	1.86E-09	
F11	avg	**0.00E00**	**0.00E00**	**0.00E00**	**0.00E00**	**0.00E00**	4.8441E-03	**0.00E00**
	std	0.00E00	0.00E00	0.00E00	0.00E00	0.00E00	6.9779E-03	0.00E00
	p-value	/	/	/	/	/	2.22E-03	/
F12	avg	1.49E00	3.45E-01	2.50E-02	6.24E-03	2.48E-01	1.38E-02	**2.18E-04**
	std	1.38E-01	4.34E-02	8.77E-03	6.11E-04	0.00E00	3.58E-02	1.74E-04
	p-value	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	2.48E-02	/
F13	avg	1.47E00	1.78E00	3.51E-01	2.19E-04	1.69E00	1.43E-03	**1.31E-10**
	std	2.30E-01	2.57E-01	1.14E-01	4.96E-04	0.00E00	4.63E-03	2.25E-10
	p-value	1.86E-09	1.86E-09	1.86E-09	3.73E-09	1.86E-09	1.86E-09	/
F14	avg	2.95E00	4.96E00	**9.98E-01**	**9.98E-01**	**9.98E-01**	**9.98E-01**	4.49E00
	std	3.06E00	4.34E00	5.91E-16	2.02E-05	1.12E-16	2.25E-16	5.44E00
	p-value	8.87E-01	2.13E-01	1.11E-02	9.84E-01	9.84E-01	2.90E-04	/
F15	avg	3.41E-04	1.02E-03	6.51E-04	5.13E-04	3.24E-04	1.48E-03	**3.07E-04**
	std	4.76E-05	4.24E-04	5.91E-04	2.68E-04	0.00E00	6.09E-04	3.92E-09
	p-value	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	1.86E-09	/
F16	avg	-1.0316284	-1.0315343	**-1.0316285**	-1.0186143	-1.0316272	**-1.0316285**	**-1.0316285**
	std	9.75E-08	9.13E-05	0.00E00	5.39E-02	2.25E-16	0.00E00	3.16E-13
	p-value	1.86E-09	1.86E-09	/	1.86E-09	1.86E-09	/	/
F17	avg	4.14E-01	4.12E-01	3.97E-01	4.75E-01	3.98E-01	**3.97E-01**	**3.97E-01**
	std	6.64E-02	1.34E-02	3.90E-08	1.13E-01	1.12E-16	5.64E-17	1.93E-10
	p-value	1.86E-09	1.86E-09	8.01E-08	1.86E-09	1.86E-09	/	/
F18	avg	3.0000004	3.005	**3.00E00**	7.25E00	3.0007E00	**3.00E00**	**3.00E00**
	std	1.31E-06	8.81E-03	1.01E-14	8.67E00	9.03E-16	9.03E-16	8.15E-15
	p-value	1.86E-09	1.86E-09	3.41E-02	1.86E-09	1.86E-09	1.03E-02	/
F19	avg	-3.85E00	-3.83E00	**-3.862E00**	-3.60E00	-3.86E00	-3.83E00	-3.86E00
	std	7.20E-03	2.27E-02	3.20E-09	2.28E-01	9.03E-16	1.41E-01	1.83E-03
	p-value	1.86E-09	1.30E-08	1.86E-09	1.86E-09	2.34E-02	7.11E-03	/
F20	avg	-2.79E00	-2.98E00	**-3.30E00**	-2.44E00	-3.08E00	-3.22E00	-3.27E00
	std	3.24E-01	7.02E-02	4.56E-02	5.30E-01	0.00E00	6.60E-02	5.01E-02
	p-value	1.86E-09	1.86E-09	4.66E-03	1.86E-09	1.86E-09	1.04E-03	
F21	avg	-1.92E00	-8.02E00	-1.01E+01	-1.01E+01	-4.96E+00	-8.89E00	**-1.015E+01**
	std	1.43E00	9.84E-01	9.72E-05	6.32E-04	1.80E-15	2.61E00	1.90E-07
	p-value	1.86E-09	1.86E-09	1.82E-05	1.86E-09	1.86E-09	1.71E-01	/
F22	avg	-2.00E00	-8.54E00	-1.02E+01	-1.04E+01	-4.94E00	-8.51E00	**-1.04E+01**
	std	1.40E00	8.29E-01	9.70E-01	2.81E-04	9.03E-16	3.22E00	1.00E-06
	p-value	1.86E-09	1.86E-09	5.97E-06	3.73E-09	1.86E-09	6.85E-01	/
F23	avg	-2.35E00	-8.56E00	-1.05E+01	-1.05360E+01	-7.14E00	-1.00E-01	**-1.05364E+01**
	std	1.46E00	9.22E-01	1.08E-04	6.59E-04	1.80E-15	1.83E00	8.65E-07
	p-value	1.86E-09	1.86E-09	2.56E-04	1.86E-09	1.86E-09	1.53E-04	/
w/t/l		0/2/21	0/6/17	3/2/18	1/5/18	0/3/20	1/5/18	/

strategy, is denoted as ADWOA.1. Subsequently, the integration of the purified search strategy into ADWOA.1 births ADWOA.2, further enriched by the inclusion of adaptive inertia weights, denoting ADWOA.3. Augmenting ADWOA.3 with Laplace

Table II

RESULTS OF ABLATION STUDY

		WOA	ADWOA.1	ADWOA.2	ADWOA.3	ADWOA.4	ADWOA
F1	best	6.13E-25	1.09E-27	2.68E-52	5.17E-317	**0**	**0**
	avg	1.28E-20	9.29E-25	3.79E-48	3.29E-306	**0**	**0**
	std	6.6137E-20	2.57E-24	1.27E-47	0	0	0
F2	best	9.00E-17	2.78E-19	3.41E-36	3.67E-172	**0**	**0**
	avg	6.56E-15	1.19E-17	6.29E-34	3.05E-165	**0**	**0**
	std	1.02E-14	2.37E-17	1.16E-33	0	0	0
F3	best	1.83E-06	2.52E-05	3.23E-07	3.14E-264	**0**	**0**
	avg	2.50E-02	3.46E-02	3.92E-02	2.06E-247	**0**	**0**
	std	1.07E-01	1.00E-01	2.10E-01	0	0	0
F4	best	2.37E-06	1.57E-04	3.18E-08	1.02E-141	**0**	**0**
	avg	2.50E-04	1.17E-03	2.20E-05	4.54E-136	6.37E-316	0
	std	5.68E-04	1.46E-03	3.76E-05	1.72E-135	**0**	**0**
F5	best	26.0762	25.3910	24.8013	27.0044	**24.0883**	24.8119
	avg	2.76E+01	27.1503	25.4411	27.6607	**24.8469**	25.0759
	std	8.68E-01	9.06E-01	3.30E-01	2.02E-01	2.61E-01	1.86E-01
F6	best	4.94E-01	3.79E-02	3.89E-03	1.66E-01	6.83E-06	**2.01E-12**
	avg	1.55E00	9.03E-01	8.31E-03	3.30E-01	2.08E-05	**2.81E-10**
	std	6.38E-01	5.67E-01	2.43E-03	8.80E-02	9.02E-06	4.00E-10
F7	best	2.77E-05	2.90E-05	7.84E-05	2.23E-06	6.35E-06	**1.86E-06**
	avg	3.71E-04	5.53E-04	8.46E-04	8.63E-05	**2.04E-05**	8.46E-05
	std	3.77E-04	6.10E-04	7.09E-04	6.76E-05	2.04E-04	8.91E-05
F8	best	-7795.8271	-8103.29925	-12129.2947	-7416.7639	-12559.0967	**-12569.4866**
	avg	-6983.9682	-6724.1753	-10227.9481	-6608.99942	-12428.5848	**-12569.4841**
	std	322.4024	503.9893	1671.6832	468.1634	147.7531	7.25E-03
F9	best	**0**	**0**	**0**	**0**	**0**	**0**
	avg	1.70E-14	4.62E-01	**0**	**0**	**0**	**0**
	std	3.04E-14	1.2486	0	0	0	0
F10	best	2.20E-13	4.30E-14	3.99E-15	**4.44E-16**	**4.44E-16**	**4.44E-16**
	avg	1.84E-11	3.27E-13	7.90E-15	**4.44E-16**	**4.44E-16**	**4.44E-16**
	std	4.71E-11	4.94E-13	2.69E-15	1.00E-31	1.002E-31	1.00E-31
F11	best	**0**	**0**	**0**	**0**	**0**	**0**
	avg	**0**	6.80E-03	**0**	**0**	**0**	**0**
	std	0	1.04E-05	0	0	0	0
F12	best	4.11E-02	2.03E-02	1.33E-04	7.75E-03	6.04E-07	**1.39E-09**
	avg	1.49E00	8.80E-02	3.72E-04	1.77E-02	**7.22E-05**	2.18E-04
	std	1.38E-01	1.08E-01	1.74E-04	6.13E-03	7.47E-04	1.74E-04

distribution perturbation yields ADWOA.4. Ultimately, the amalgamation of all strategies culminates in the comprehensive ADWOA framework posited in this study.

These distinct algorithmic configurations are subjected to meticulous ablation experiments employing benchmark test functions, with the empirical findings meticulously documented in Table VI. Analysis of the outcomes underscore ADWOA.1's noteworthy enhancements across functions F_1, F_2, F_6, F_{10}, F_{22}, and F_{23} vs the classical WOA, albeit its suboptimal performance stemming from the absence of the remaining optimization strategies. Noteworthy advancements are observed upon the incorporation of the purified search strategy, underscoring performance enhancements over WOA and ADWOA.1, albeit exhibiting minutely inferior results in functions F_7, F_{16}, and F_{17} than the classical WOA. This discrepancy signals the efficacy of the interdimensional information exchange within the purified search strategy in fortifying local fine search and global exploration capacities.

C. Stability Analysis

To evaluate the distribution characteristics of ADWOA across 23 benchmark functions, a statistical analysis was conducted using the results of 30 independent simulation runs. Box-and-whisker plots were used to visualize the distribution of the solutions, including the median, quartiles, and potential outliers. This visualization provides a clear basis for comparing ADWOA with other algorithms.

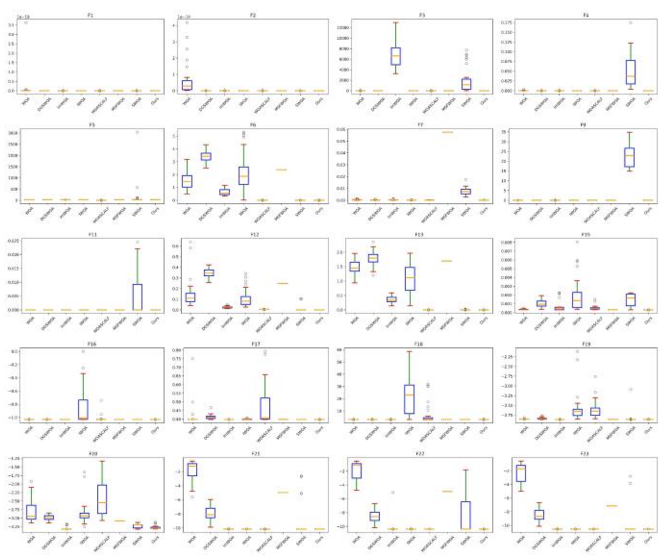

Fig. 2. Box line diagram of ADWOA with improved WOA

Fig. 2 and Fig. 3 present the distribution of ADWOA's results in comparison with several WOA variants and other evolutionary algorithms. Across most functions, ADWOA shows strong stability and competitive performance, with lower medians, quartiles, and overall value ranges relative to other methods. These observations indicate that ADWOA maintains stable behavior and produces high-quality solutions on a wide set of functions. Some variations in distribution width appear for certain functions. These differences mainly result from the algorithm achieving higher accuracy in those cases, reflecting its adaptive search behavior. While ADWOA demonstrates broad exploration capability, the observed variability suggests that further refinement may improve consistency, providing a direction for future work.

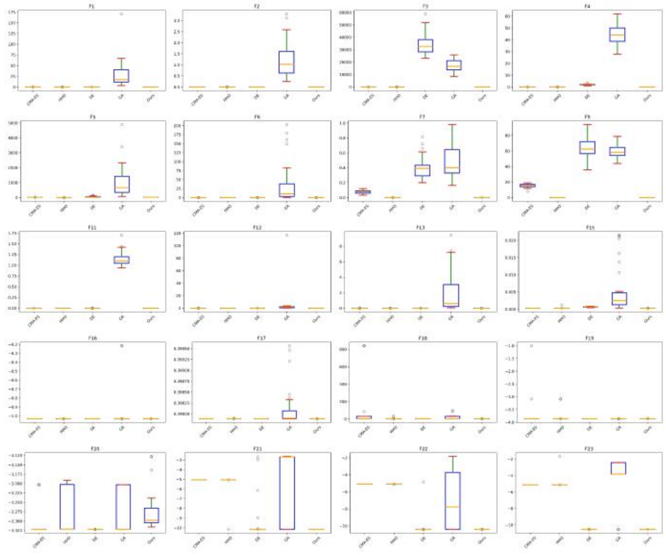

Fig. 3. Box line diagram of ADWOA with evolutionary algorithms

V. Conclusions

Aiming to address the shortcomings of the Whale Optimization Algorithm (WOA), such as its tendency to fall into local optima and its inability to balance search and exploitation, this paper proposes an adaptive differential whale optimization algorithm. ADWOA is based on WOA and introduces five strategies. Firstly, dynamic convergence factors and adaptive inertia weights enhance search and exploitation abilities. Secondly, the purified search idea exchanges information between different dimensions. Thirdly, Laplace distribution perturbation is introduced to increase population diversity and improve local search ability. Lastly, it is combined with the differential evolutionary algorithm to enhance performance further. Experimental and statistical results from 23 benchmark functions show that ADWOA has superior optimization ability and stability compared to other algorithms in solving global optimization problems.

In future work, modifications to ADWOA's search and development strategy will be considered to address its shortcomings in general stability, slow convergence speed, and low convergence accuracy on individual test functions. Combining it with other meta-heuristic algorithms will also be explored to improve its performance further. Additionally, its applicability and feasibility can be extended to complex optimization problems such as weather load prediction, neural network dynamic parameter selection, and robot path planning.

References

[1]. R. J. Kuo and F. E. Zulvia, "The gradient evolution algorithm: A new metaheuristic," Information Sciences, vol. 316, pp. 246–265, 2015.

[2]. B. T. Polyak, "Newton's method and its use in optimization," European Journal of Operational Research, vol. 181, no. 3, pp. 1086–1096, 2007.

[3]. J. Nocedal and S. J. Wright, "Quasi-newton methods," in Numerical Optimization, New York: Springer, 2006, pp. 135–163.

[4]. J. Nocedal and S. J. Wright, "Conjugate gradient methods," in Numerical Optimization, New York: Springer, 2006, pp. 101–134.

[5]. H. P. Gavin and J. T. Scruggs, "Constrained optimization using lagrange multipliers," CEE 201L, 2012.

[6]. C. Blum, "Ant colony optimization: Introduction and recent trends," Physics of Life Reviews, vol. 2, no. 4, pp. 353–373, 2005.

[7]. R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in IEEE International Symposium on Micro Machine and Human Science, 1995, pp. 39–43.

[8]. X. L. Li, "An optimizing method based on autonomous animats: fish-swarm algorithm," Systems Engineering-Theory & Practice, vol. 22, no. 11, pp. 32–38, 2002.

[9]. X. S. Yang, "Firefly algorithm, Levy flights and global optimization," in Research and Development in Intelligent Systems XXVI: Incorporating Applications and Innovations in Intelligent Systems XVII, London, U.K.: Springer, 2010, pp. 209–218.

[10]. X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," in IEEE World Congress on Nature & Biologically Inspired Computing, 2009, pp. 210–214.

[11]. S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in Engineering Software, vol. 69, pp. 46–61, 2014.

[12]. W. T. Pan, "A new fruit fly optimization algorithm: taking the financial distress model as an example," Knowledge-Based Systems, vol. 26, pp. 69–74, 2012.

[13]. S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, et al., "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," Advances in Engineering Software, vol. 114, pp. 163–191, 2017.

[14]. A. A. Heidari, S. Mirjalili, H. Faris, et al., "Harris hawks optimization: Algorithm and applications," Future Generation Computer Systems, vol. 97, pp. 849–872, 2019.

[15]. S. Li, H. Chen, M. Wang, et al., "Slime mould algorithm: A new method for stochastic optimization," Future Generation Computer Systems, vol. 111, pp. 300–323, 2020.

[16]. L. Abualigah et al., "Aquila optimizer: a novel meta-heuristic optimization algorithm," Computers & Industrial Engineering, vol. 157, p. 107250, 2021.

[17]. W. Zhao, L. Wang, and S. Mirjalili, "Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications," Computer Methods in Applied Mechanics and Engineering, vol. 388, p. 114194, 2022.

[18]. M. A. Basset, R. Mohamed, M. Jameel, et al., "Nutcracker optimizer: A novel nature-inspired metaheuristic algorithm for global optimization and engineering design problems," Knowledge-Based Systems, vol. 262, p. 110248, 2023.

[19]. M. A. Basset, R. Mohamed, and M. Abouhawwash, "Crested porcupine optimizer: a new nature-inspired metaheuristic," Knowledge-Based Systems, vol. 284, p. 111257, 2024.

[20]. S. Mirjalili and A. Lewis, "The whale optimization algorithm," Advances in Engineering Software, vol. 95, pp. 51–67, 2016.

[21]. J. Rahebi, "Vector quantization using whale optimization algorithm for digital image compression," Multimedia Tools and Applications, vol. 81, no. 14, pp. 20077–20103, 2022.

[22]. G. Kaur and S. Arora, "Chaotic whale optimization algorithm," Journal of Computational Design and Engineering, vol. 5, no. 3, pp. 275–284, 2018.

[23]. Y. Feng, H. Chen, T. Li, et al., "A novel community detection method based on whale optimization algorithm with evolutionary population," Applied Intelligence, vol. 50, no. 8, pp. 2503–2522, 2020.

[24]. G. Sun, Y. Shang, K. Yuan, et al., "An improved whale optimization algorithm based on nonlinear parameters and feedback mechanism," International Journal of Computational Intelligence Systems, vol. 15, no. 1, p. 38, 2022.

[25]. S. Chakraborty, A. K. Saha, S. Sharma, et al., "A novel enhanced whale optimization algorithm for global optimization," Computers & Industrial Engineering, vol. 153, p. 107086, 2021.

[26]. A. Elmogy, H. Miqrish, W. Elawady, et al., "ANWOA: an adaptive nonlinear whale optimization algorithm for high-dimensional

optimization problems," Neural Computing and Applications, vol. 35, no. 30, pp. 22671–22686, 2023.
[27]. J. Zhang and J. S. Wang, "Improved whale optimization algorithm based on nonlinear adaptive weight and golden sine operator," IEEE Access, vol. 8, pp. 77013–77048, 2020.
[28]. S. Chakraborty, A. K. Saha, S. Sharma, et al., "A hybrid whale optimization algorithm for global optimization," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 1, pp. 431–467, 2023.
[29]. W. Gao, S. Liu, and L. Huang, "A global best artificial bee colony algorithm for global optimization," Journal of Computational and Applied Mathematics, vol. 236, no. 11, pp. 2741–2753, 2012.
[30]. Q. Fan, Z. Chen, Z. Li, et al., "A new improved whale optimization algorithm with joint search mechanisms for high-dimensional global optimization problems," Engineering with Computers, vol. 37, pp. 1851–1878, 2021.
[31]. M. Mohammadi, S. Farzin, S. F. Mousavi, et al., "Investigation of a new hybrid optimization algorithm performance in the optimal operation of multi-reservoir benchmark systems," Water Resources Management, vol. 33, pp. 4767–4782, 2019.
[32]. Y. Che and D. He, "A hybrid whale optimization with seagull algorithm for global optimization problems," Mathematical Problems in Engineering, vol. 2021, pp. 1–31, 2021.
[33]. D. Prabhakar and M. Satyanarayana, "Side lobe pattern synthesis using hybrid SSWOA algorithm for conformal antenna array," Engineering Science and Technology, vol. 22, no. 6, pp. 1169–1174, 2019.
[34]. D. Oliva, M. A. E. Aziz, and A. E. Hassanien, "Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm," Applied Energy, vol. 200, pp. 141–154, 2017.
[35]. A. Singh, A. Khamparia, and F. A. Turjman, "A hybrid evolutionary approach for multi-objective unit commitment problem in power systems," Energy Reports, vol. 11, pp. 2439–2449, 2024.
[36]. B. Yin, C. Wang, and F. Abza, "New brain tumor classification method based on an improved version of whale optimization algorithm," Biomedical Signal Processing and Control, vol. 56, p. 101728, 2020.
[37]. Y. Zhang, Y. Liu, J. Li, et al., "WOCDA: A whale optimization based community detection algorithm," Physica A: Statistical Mechanics and its Applications, vol. 539, p. 122937, 2020.
[38]. J. Anitha, S. I. A. Pandian, and S. A. Agnes, "An efficient multilevel color image thresholding based on modified whale optimization algorithm," Expert Systems with Applications, vol. 178, p. 115003, 2021.
[39]. G. Ning, "Two-dimensional Otsu multi-threshold image segmentation based on hybrid whale optimization algorithm," Multimedia Tools and Applications, vol. 82, no. 10, pp. 15007–15026, 2023.
[40]. Y. Dai, J. Yu, C. Zhang, et al., "A novel whale optimization algorithm of path planning strategy for mobile robots," Applied Intelligence, vol. 53, no. 9, pp. 10843–10857, 2023.
[41]. Y. Huang, Y. Li, Z. Zhang, et al., "A novel path planning approach for AUV based on improved whale optimization algorithm using segment learning and adaptive operator selection," Ocean Engineering, vol. 280, p. 114591, 2023.
[42]. M. Pant, H. Zaheer, L. G. Hernandez, et al., "Differential Evolution: A review of more than two decades of research," Engineering Applications of Artificial Intelligence, vol. 90, p. 103479, 2020.
[43]. Y. Zhang, D. Gong, X. Gao, et al., "Binary differential evolution with self-learning for multi-objective feature selection," Information Sciences, vol. 507, pp. 67–85, 2020.
[44]. B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems," *Computers & Industrial Engineering*, vol. 158, p. 107408, 2021.
[45]. Z. Liang, T. Shu, and Z. Ding, "A novel improved whale optimization algorithm for global optimization and engineering applications," Mathematics, vol. 12, no. 5, p. 636, 2024.
[46]. S. Chakraborty, S. Sharma, A. K. Saha, et al., "A novel improved whale optimization algorithm to solve numerical optimization and real-world applications," Artificial Intelligence Review, pp. 1–112, 2022.
[47]. A. Seyyedabbasi, "WOASCALF: A new hybrid whale optimization algorithm based on sine cosine algorithm and levy flight to solve global optimization problems," Advances in Engineering Software, vol. 173, p. 103272, 2022.
[48]. W. Wei, L. Mengshan, W. Yan, et al., "Cluster energy prediction based on multiple strategy fusion whale optimization algorithm and light gradient boosting machine," BMC chemistry, vol. 18, no. 1, p. 24, 2024.
[49]. L. Wu, E. Chen, Q. Guo, et al., "Smooth exploration system: A novel ease-of-use and specialized module for improving exploration of whale optimization algorithm," Knowledge-Based Systems, vol. 272, p. 110580, 2023.