SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I10P105

Volume 11 Issue 10, 47-52, October 2024
© 2024 Seventh Sense Research Group®

Original Article

An Efficient Global Optimization-Based Grey Wolf Optimization Algorithm for Fast Convergence to Various Optimization Functions

Dipak Patel[1], Bhavesh Patel[2], Prashant Modi[3], Shakti Patel[4], Rahul Shah[5]

[1]*Information Technology Department, VGEC, Gujarat, India.*
[2]*Computer Science and Engineering Department, GEC Patan, Gujarat, India.*
[3]*Information Technology Department, GEC Modasa, Gujarat, India.*
[4]*Computer Engineering Department, GP Gandhinagar, Gujarat, India.*
[5]*Computer Engineering Department, VGEC, Gujarat, India.*

[1]*Corresponding Author : dipak.eng@gmail.com*

Received: 03 August 2024 Revised: 04 September 2024 Accepted: 05 October 2024 Published: 30 October 2024

Abstract - *The Grey Wolf Optimizer (GWO), a bio-inspired metaheuristic algorithm, has gained prominence for solving complex optimization problems across various domains. Despite its advantages, the standard GWO algorithm often suffers from premature convergence and inefficacy in handling local optima, limiting its applicability for global optimization tasks. This research paper introduces an efficient GWO algorithm incorporating a novel adaptive search mechanism designed to overcome local optima entrapment issues and slow convergence rates inherent in the conventional GWO approach. This research analyses the behavior of the traditional GWO algorithm and identifies its key limitations in the exploration and exploitation phases. Then, a modified exploration technique is proposed with an adaptive exploitation method, dynamically adjusting the position update mechanism of wolves. The proposed modifications aim to sustain diversity in the search space and enhance the global search capability, thus accelerating convergence towards the global optimum and converging fast to the solution. Extensive experimental evaluations on several benchmark unimodal and multimodal functions demonstrate that the modified GWO algorithm significantly outperforms the original version and other contemporary optimization techniques regarding convergence speed, solution accuracy, and robustness against local optima. This research not only presents a viable solution to the limitations of the standard GWO but also contributes to the broader field of swarm intelligence, offering insights that could inspire further innovations in metaheuristic algorithms.*

Keywords - *Global optima, Grey Wolf Optimization (GWO), Metaheuristic algorithm, Optimization problems, Swarm intelligence.*

1. Introduction

The Grey Wolf Optimization (GWO) algorithm [1], based on grey wolves' social hierarchy and hunting activity, has emerged as a powerful metaheuristic optimization technique for solving complex optimization problems across various domains. GWO mimics the collaborative hunting strategy of grey wolves, where individuals coordinate their movements to locate and capture prey effectively. Despite its initial success and widespread adoption, the standard GWO algorithm exhibits certain limitations that hinder its performance in tackling real-world optimization problems.

One of the primary challenges the conventional GWO algorithm faces is its susceptibility to premature convergence and inefficient solution space exploration [2, 3]. The standard updates equations for the position of wolves lack adaptability and may lead to stagnation in local optima, particularly in high-dimensional and multimodal optimization landscapes. Additionally, the slow convergence rate of the naive GWO algorithm can impede its effectiveness in finding globally optimal solutions within a reasonable computational budget.

To address these shortcomings and enhance the efficacy of the GWO algorithm [4], this research proposes modifications to the updated position equations of wolves. By introducing novel strategies for adjusting the movement of wolves during the optimization process, we aim to mitigate the issues of premature convergence and limited exploration capacity inherent in the conventional GWO approach.

The proposed modifications seek to strike a balance between exploration and exploitation, which is crucial for

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

effective optimization in diverse and challenging problem domains. By incorporating adaptive mechanisms and dynamic adjustments to the update equations, the modified GWO algorithm aims to enhance its ability to escape local optima traps and converge more efficiently towards globally optimal solutions.

This paper presents a detailed analysis of the limitations of the standard GWO algorithm and elucidates the rationale behind the proposed modifications. A comprehensive overview of the revised update equations is provided, and discussed their theoretical underpinnings are discussed in the context of enhancing exploration-exploitation trade-offs. Furthermore, extensive empirical evaluations on benchmark functions are conducted to assess the performance and efficacy of the modified GWO algorithm compared to its conventional counterpart and other state-of-the-art optimization techniques.

This research endeavor contributed to advancing metaheuristic optimization algorithms by providing insights into designing more robust and efficient optimization techniques. The proposed modifications to the GWO algorithm hold the potential to broaden its applicability and effectiveness in solving complex optimization problems with improved convergence rates and solution quality.

2. Contribution Made in the Paper

Introduced an Efficient Grey Wolf Optimization (Eff. GWO) algorithm to overcome the limitations like premature convergence and ineffective handling of local optima by standard GWO. The proposed modifications focus on an adaptive exploration and exploitation mechanism, improving global search capability and accelerating convergence. Empirical evaluations on benchmark functions (Sphere and Rastrigin) demonstrate that Eff. GWO outperforms traditional GWO and ex-GWO regarding convergence speed and solution accuracy.

The study contributes to the broader field of swarm intelligence by offering a more robust and computationally efficient metaheuristic algorithm.

3. Literature Survey

Metaheuristic algorithms have gained much attention in finding solutions to optimization problems. Among them, the Grey Wolf Optimizer (GWO), proposed by Mirjalili et al. [1], stands out due to its ability to mimic grey wolves' leadership hierarchy and hunting strategies. Despite its initial success in optimization tasks, the standard GWO exhibits limitations, such as premature convergence and trapping to local optima, especially in high-dimensional and multimodal functions.

Further, Li et al. [3] introduced an improved version of GWO tailored to solve engineering optimization problems. Modifying the position update equations aimed to improve GWO’s ability to escape local optima and enhance its global search capabilities. Their findings showed that the improved GWO variant outperformed the standard algorithm regarding convergence rate and solution accuracy across various engineering problems.

In the context of structural optimization, Upadhyay et al. [4] conducted a comparative study of standard, modified, and variable weight GWO for 2D structural shape optimization. Their research highlighted the limitations of standard GWO in handling local optima and demonstrated the superiority of modified versions in terms of convergence speed and robustness. The application of GWO in numerical optimization problems has also been explored by Mohammed et al. [5], who introduced enhancements to GWO to handle complex, nonlinear objective functions better. Their research showcased the effectiveness of GWO variants in solving both unimodal and multimodal functions, particularly when augmented with adaptive mechanisms that dynamically adjust the exploration and exploitation phases.

Another notable modification is the Ex-GWO introduced by Seyyedabbasi and Kiani [6]. This algorithm improved the position update mechanism by considering the positions of all wolves in the previous iterations rather than focusing only on the alpha, beta, and delta wolves. The experimental results proved that Ex-GWO outperformed the standard GWO in terms of convergence speed and solution accuracy, although the proposed algorithm had the cost of additional computational overhead.

Given the limitations identified in the literature, the present work introduces an Efficient Grey Wolf Optimization (Eff. GWO) algorithm to overcome premature convergence and slow exploitation in the standard GWO. By modifying the position update mechanism and introducing adaptive exploitation strategies, this research aims to improve GWO's computation time and accuracy performance on benchmark functions such as Sphere and Rastrigin. Comparative analysis with Ex-GWO and other variants further validates the superiority of the proposed Eff. GWO.

4. Materials and Methods

4.1. Grey Wolf Optimizer

The GWO algorithm [1] is a meta-heuristic algorithm proposed and employed to solve various optimization-based functions [7]. The algorithm is based on grey wolves' social hierarchy and hunting activity, which exhibit cooperative strategies for efficient prey localization. GWO mimics the hierarchical structure of a wolf pack, as shown in Figure 1, comprising alpha, beta, delta, and omega wolves, representing the dominant, subdominant, subordinate, and omega wolves, respectively. The alpha wolf represents the dominant individual in the wolf pack, exhibiting leadership and authority. In the context of GWO, the alpha wolf corresponds to the best-performing solution found thus far in the optimization process. The position of the alpha wolf serves as a reference for other wolves, guiding their movement towards promising regions of the search space. The update equation for the alpha wolf involves a balance between exploration and exploitation, aiming to refine the search around the best-known solution. The beta wolf is the second-highest-ranking individual in the wolf pack, supporting the alpha wolf and assisting in leading the pack.

In GWO, the beta wolf represents a solution that is inferior to the alpha wolf but still competitive. The beta wolf's position influences other wolves' movement, providing additional guidance towards promising areas of the search space. The delta wolf is subordinate to alpha and beta wolves but holds a higher rank than the omega wolf.

In GWO, the delta wolf represents a solution inferior to the alpha and beta wolves but still of moderate quality. The updated equation for the delta wolf emphasizes exploration, encouraging movement towards unexplored areas while exploiting promising solutions. The omega wolf is the lowest-ranking individual in the wolf pack, often deferring to the top wolves, alpha, beta, and delta. While the omega wolf may not contribute significantly to the optimization process, its presence ensures diversity in the search population.

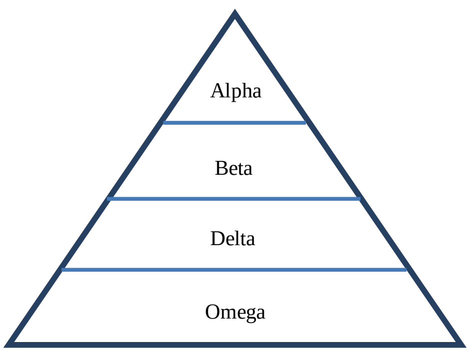

Fig. 1 Hierarchy of wolves

The attacking activity of grey wolves is divided into three parts: first, tracking of prey is done, then harassing the prey, and finally, the attack on the prey is carried out. Based on the dimensionality of the data, the number of elements in the position vector of a wolf is determined.

Let vector $J_i = [j_1, j_2, \ldots\ldots j_k]$ indicate the place of the i^{th} number wolf. The equations used to change the position of a wolf are shown below:

$$\vec{Z} = |\vec{L}.\vec{J_p}(t) - \vec{J}(t)| \quad (1)$$

$$\vec{J}(t+1) = \vec{J_p}(i) - \vec{G}.\vec{Z} \quad (2)$$

In the above equations, t indicates the current iteration, and t+1 represents the next iteration. Vector $\vec{J_p}$ shows the position vector of the prey and the vector $\vec{J}$ indicates the current position of the wolf. $\vec{L}$ and $\vec{G}$ are vectors which are computed according to the following equations:

$$\vec{G} = 2.\vec{a}.\vec{r_1} - \vec{a} \quad (3)$$

$$\vec{L} = 2.\vec{r_2} \quad (4)$$

Here, the vector is reduced from two to zero in entire iterations. Here, vectors $\vec{r_1}$ and $\vec{r_2}$ specifies any random vectors in the range [0, 1].

4.2. Hunting the Prey

Gray wolves possess the ability to detect the location of potential prey, with the search process primarily guided by the alpha (α), beta (β), and delta (δ) wolves. In all iterations, the top three wolves α, β, and δ in the current iteration are reserved, and the positions of the other wolves are updated based on the information from these leading wolves. The following formulas are used for this purpose.

$$\vec{J_1} = \vec{J_\alpha} - \vec{G_1}.|\vec{L_1}.\vec{J_\alpha} - \vec{J}| \quad (5)$$

$$\vec{J_2} = \vec{J_\beta} - \vec{G_2}.|\vec{L_2}.\vec{J_\beta} - \vec{J}| \quad (6)$$

$$\vec{J_3} = \vec{J_\delta} - \vec{G_3}.|\vec{L_3}.\vec{J_\delta} - \vec{J}| \quad (7)$$

$$\vec{J}(i+1) = \frac{\vec{J_1}+\vec{J_2}+\vec{J_3}}{3} \quad (8)$$

In the above equations, $\vec{J_\alpha}$, $\vec{J_\beta}$, and $\vec{J_\delta}$ indicate the position vector of alpha (α), beta (β), and delta (δ) wolves respectively. The position updating mechanism is visualized in Figure 2.

4.3. Efficient GWO

Although the traditional GWO [1] approach offers many benefits, it also faces certain limitations like susceptibility to local optima and slow convergence. To overcome these limitations, enhancements have been introduced, resulting in a new and efficient model called Eff. GWO. In the efficient GWO, the proposed modification occurs during the wolves' final position update formulation. Traditionally, GWO updates the position based on the current positions of α, β, and δ wolves, as per Equation (8). However, in the modified proposed approach, the next position of the wolf, J(t+1), also considers the existing position (J_k) [8] as given in Equation (9). Consequently, the formula of J_k is given in Equation (10), where the immediate best and worst solutions are named J^+ and J^- respectively. This modification results in finding the solution in less number of iterations and increases the accuracy, as discussed in the results section.

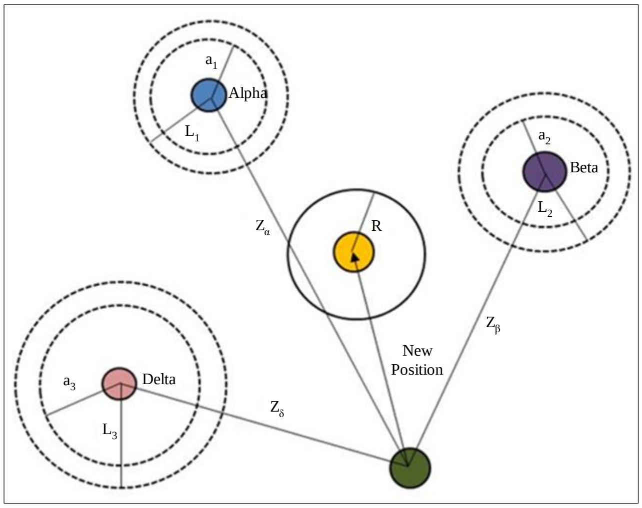

Fig. 2 Position updating in GWO

$$\vec{J}(t+1) = \frac{\vec{J_1}+\vec{J_2}+\vec{J_3}+\vec{J_k}}{4} \quad (9)$$

$$\vec{J_k} = \vec{J} + \vec{r_1}(J^+ - J^-) + \vec{r_2}(J^+ - J^-) \quad (10)$$

The pseudo-code of the GWO algorithm is presented below.

GWO Algorithm:
- Assign random values to the initial population of wolves
- Calculate the fitness of each wolf.
- Initialize top wolves as alpha, beta, delta, and omega wolves.
- While (End condition not arrived) do
  - For each wolf, do
    - Change position using Equation (8)
    - Apply boundary constraints if necessary
  - End for
  - Update vectors (a) $^{\rightarrow}$, (G) $^{\rightarrow}$and (L) $^{\rightarrow}$
  - Evaluate fitness of new position
  - Update alpha, beta, delta, and omega wolves
- End while

Eff. GWO Algorithm:
- Initialization of wolves
- Calculate the quality value of complete search agents
- Set, as first three best search agent
- While (End condition not arrived)
  - For each wolf
  - Change the position of wolves as per Equation (9)
  - End of for
  - Change parameters (a) $^{\rightarrow}$, (G) $^{\rightarrow}$and (L) $^{\rightarrow}$
  - Calculate the fitness of complete search agents
  - Change the position of wolves, and
  - Next iteration t = t + 1
- End of while
- Return (Best Wolf)

5. Simulation Results and Comparison

The superiority of the Eff. GWO algorithm is evaluated using two benchmark functions (unimodal and multimodal), as shown in Table 1, the comparison of Eff. GWO is also made with the other variant of GWO, the Expanded-GWO (Ex-GWO) [6]. Seyyedabbasi et al. [6] modified the position update Equation (8) of wolves as below:

$$J_n(t+1) = \frac{1}{n\text{-}1}\sum_{i=1}^{n-1} J_i(t); \quad n = 4,5,6,\ldots..m \quad (11)$$

In the above equation, symbol n represents the current wolf, m indicates the number of wolves in the pack, and t represents the current iteration; parameter i selects the wolf from one to the last wolf. In this version of GWO, the nth wolf updates the position from previous n-1 wolves.

Table 1. Benchmark functions used to test Eff. GWO

Function Name	Function Formula	Range	Optima	Type
Sphere	$F1(x) = \sum_{1=1}^{n} x_i^2$	[-10, 10]	0	Unimodal
Rastrigin	$F2(x) = \sum_{1=1}^{n} [x_i^2 - 10 * \cos(2\Pi x_i) + 10]$	[-5.12, 5.12]	0	Multimodal

Table 1 shows two benchmark functions, namely sphere and Rastrigin functions, which are used to test variants of GWO algorithms. Benchmark functions [7, 9, 10] shown in the above table are used to test the efficiency or ability of an algorithm to find the optimal value.

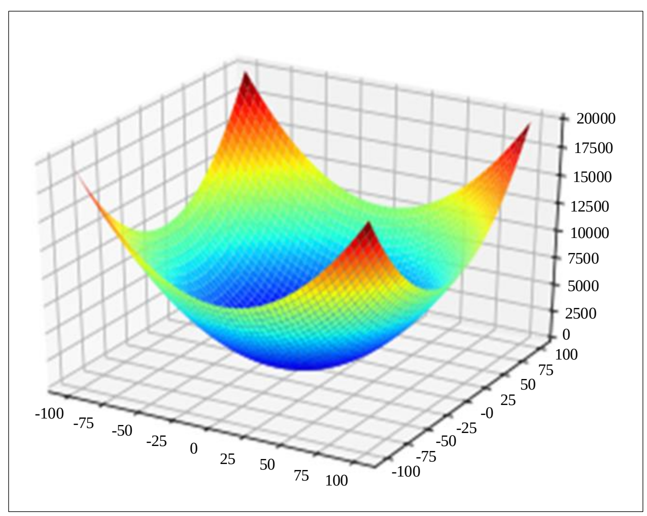

Fig. 3 Unimodal square function

The sphere function is an unimodal function with one global and local minima, whereas the multimodal function has multiple local minima. Figure 3 shows a 3-D visualization of Sphere and Rastrigin, respectively, for better understanding.

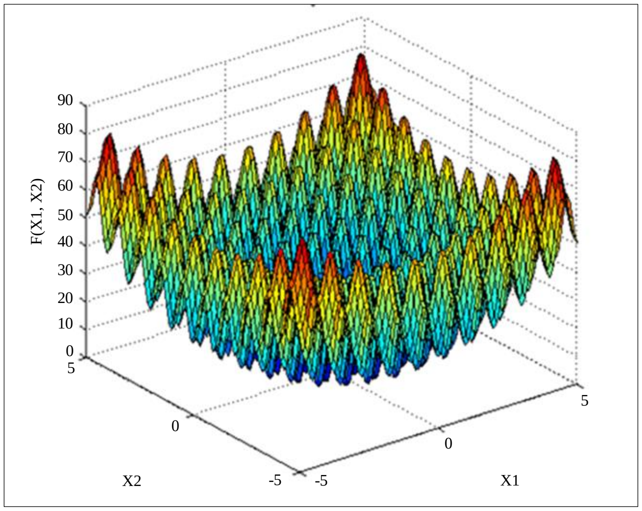

Fig. 4 Multimodal rastrigin function

To compare both variants of GWO, the solution search was conducted in the range of -10 to 10, with an initial population set to 50. All populations were assigned three-dimensional random values. $\vec{r_1}$ and $\vec{r_2}$ Equations (3) and (4) were assigned random values in the range of [0 1], and a value decreases linearly from 2 to 0. Maximum number of iterations was set to 200.

Table 2 illustrates that Eff. GWO achieved the optimal value for the sphere function within 20 iterations. In contrast, another variant of GWO (Ex-GWO) failed to reach the optimal value even after 120 iterations, obtaining a value of 14.603, which is significantly far from optimal. Additionally, GWO required 5 more iterations than Eff. GWO to attain the optimal value.

Table 3 compares Ex-GWO and Eff. GWO on the Rastrigin function to find the optimal value of 0. Eff. GWO achieved the optimal value within 90 iterations, while Ex-GWO did not reach the optimal value even after 120 iterations, obtaining a value of 0.016. Additionally, GWO required 125 iterations to reach the optimal value, indicating that Eff. GWO converges faster than GWO.

Furthermore, Ex-GWO has an additional overhead of $O(n^2)$ time complexity because it considers the positions of all previous wolves to update the position of the next wolf. On the other hand, efficient GWO has no extra overhead, and its time complexity is linear. The above comparison is also shown in the below chart for better visualization.

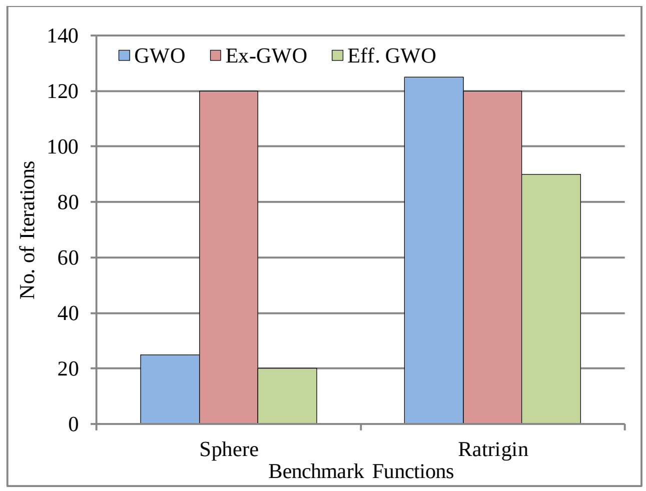

Fig. 5 Comparative chart for benchmark functions

Table 2. Comparison of Ex-GWO and Eff. GWO on sphere function

Algorithm	Optima for Sphere Function	No. of Maximum iterations Took	The Value Obtained at the End of Maximum Iterations	Time Complexity
Ex-GWO	0	120	14.603	The extra overhead of n^2 (n is no. of wolves)
Eff. GWO	0	20	0	No extra overhead

Table 3. Comparison of Ex-GWO and Eff. GWO on rastrigin function

Algorithm	Optima for Rastrigin Function	No. of Maximum Iterations Took	The Value obtained after the Maximum Number of Iterations	Time Complexity
Ex-GWO	0	120	0.016	The extra overhead of n^2 (n is no. of wolves)
Eff. GWO	0	90	0.00	No extra overhead

6. Conclusion

This research paper presents the efficient version of the GWO algorithm to overcome the problems of local optima and slow convergence to the solution of the basic GWO algorithm. Empirically testing of Eff. GWO algorithm is performed on unimodal and multimodal benchmark functions, namely Sphere and Rastrigin. The empirical results demonstrate the superiority of Eff. GWO algorithm is used to solve standard benchmark functions and fast convergence of solutions on unimodal and multimodal functions specifically, Eff. GWO consistently achieved the optimal values significantly faster than both Ex-GWO and the standard GWO. For the sphere function, Eff. GWO reached the optimal value within 20 iterations, whereas Ex-GWO failed to do so even after 120 iterations, obtaining a value far from optimal.

Similarly, for the Rastrigin function, Eff. GWO achieved the optimal value in 90 iterations, while Ex-GWO could not achieve the optimal value even after 120 iterations, and standard GWO took 125 iterations to converge means Eff. GWO can reach the optimal value in the case of the Rastrigin function with 28% lower iterations compared to basic GWO—the Eff. GWO algorithm outperforms the other GWO variants regarding convergence speed, solution accuracy, and computational efficiency. These findings validate the modifications introduced in Eff. GWO and underscore its potential as a robust and efficient optimization tool for solving complex optimization problems. Future research can extend the application of Eff. GWO to other domains and explore further enhancements to improve its performance.

References

[1] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46-61, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[2] Manoharan Premkumar et al., "Augmented Weighted K-Means Grey Wolf Optimizer: An Enhanced Metaheuristic Algorithm for Data Clustering Problems," *Scientific Reports*, vol. 14, no. 1, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[3] Yu Li, Xiaoxiao Lin, and Jingsen Liu, "An Improved Gray Wolf Optimization Algorithm to Solve Engineering Problems," *Sustainability*, vol. 13, no. 6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Bhavik D. Upadhyay, Sunil S. Sonigra, and Sachin D. Daxini, "A Comparative Study of Standard, Modified and Variable Weight Grey Wolf Optimizer for 2D Structural Shape Optimization," *Iranian Journal of Science and Technology, Transactions of Mechanical Engineering*, vol. 47, no. 4, pp. 1957-1984, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Hardi Mohammed, Zrar Abdul, and Zana Hamad, "Enhancement of GWO for Solving Numerical Functions and Engineering Problems," *Neural Computing and Applications*, vol. 36, no. 7, pp. 3405-3413, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Amir Seyyedabbasi, and Farzad Kiani, "I-GWO and Ex-GWO: Improved Algorithms of the Grey Wolf Optimizer to Solve Global Optimization Problems," *Engineering with Computers*, vol. 37, no. 1, pp. 509-532, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Bilal H. Abed-alguni, and Noor Aldeen Alawad, "Distributed Grey Wolf Optimizer for Numerical Optimization Problems," *Applied Soft Computing*, vol. 102, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Zhou Wu, and Tommy W.S. Chow, "Neighborhood Field for Cooperative Optimization," *Soft Computing*, vol. 17, no. 5, pp. 819-834, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[9] J.J. Liang, B.Y. Qu, and P.N. Suganthan, "*Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization*," Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013. [Google Scholar] [Publisher Link]

[10] J. J. Liang et al., "*Problem Definitions and Evaluation Criteria for the CEC 2015 Competition on Learning-Based Real-Parameter Single Objective Optimization*," Technical Report201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2014. [Google Scholar] [Publisher Link]