



ISSN: 0067-2904

Grey Wolf Optimization Algorithm: A Survey

Mohammed H. Hashem*, Hasanen S. Abdullah, Khalil I. Ghathwan

Department of Computer sciences, University of Technology, Baghdad, Iraq

Received: 1/6/2022

Accepted: 1/1/2023

Published: 30/11/2023

Abstract

The Gray Wolf Optimizer (GWO) is a population-based meta-heuristic algorithm that belongs to the family of swarm intelligence algorithms inspired by the social behavior of gray wolves, in particular the social hierarchy and hunting mechanism. Because of its simplicity, flexibility, and few parameters to be tuned, it has been applied to a wide range of optimization problems. And yet it has some disadvantages, such as poor exploration skills, stagnation at local optima, and slow convergence speed. Therefore, different variants of GWO have been proposed and developed to address these disadvantages. In this article, some literature, especially from the last five years, has been reviewed and summarized by well-known publishers. First, the inspiration and the mathematical model of GWO were explained. Subsequently, the improved GWO variants were divided into four categories and discussed. After that, each variant's methodology and experiments were explained and clarified. The study ends with a summary conclusion of the main foundation of GWO and suggests some possible future directions that can be explored further.

Keywords: Grey wolf optimizer, optimization, metaheuristic, GWO variants, GWO applications.

خوارزمية محسن الذئب الرمادي: مراجعة

محمد حميد هاشم*، حسنين سمير عبدالله، خليل ابراهيم عثمان

قسم علوم الحاسوب، الجامعة التكنولوجية، العراق، بغداد

الخلاصة

محسن الذئب الرمادي هي خوارزمية ما بعد الحدس قائمه على السكان تنتمي الى مجموعة خوارزميات ذكاء السرب استلهمت من السلوك الاجتماعي وآلية الصيد في الذئاب الرمادية في الطبيعة. نظراً لبساطتها ومرونتها والعديد القليل من المعلومات المطلوب ضبطها تم تطبيقها في العديد من مشاكل التحسين. ومع ذلك تعاني من بعض أوجه القصور مثل الضعف في قابليات الاستكشاف والركود في الحل الأمثل المحلي وسرعة التقارب البطيئة. لذلك تم اقتراح وتطوير العديد من النسخ المحسنة من الخوارزمية من اجل التغلب على المساوئ المذكورة. في هذه الدراسة تم استعراض وتلخيص بعض الدراسات، خاصة في السنين الخمسة الماضية المنشورة في دور النشر المعروفة. أولاً تم توضيح إلهام الخوارزمية ونموذجها الرياضي، ومن ثم تم تقسيم النسخ المحسنة والمقترحة في الدراسات الى أربع اقسام وشرحها، بعد ذلك تم توضيح منهجيات وتجارب كل نسخة محسنة من الخوارزمية. تنتهي الدراسة الحالية بخلاصة موجزة للأساس الرئيسي للخوارزمية وتقتصر بعض الاعمال المستقبلية التي يمكن استكشافها بشكل أكبر.

*Email: cs.20.67@grad.uotechnology.edu.iq

1. Introduction

The process of maximizing or minimizing some criterion such as cost or time consumption in a certain system is referred to as optimization. The complexity of the optimization problem increases with the number of problem variables to be optimized.

Among the various optimization techniques utilized, meta-heuristics are particularly popular in the field of optimization and have been employed in a variety of fields, including engineering, science, economics, administration, and commerce. Meta-heuristic algorithms are used to find the best near-optimal solution within a reasonable timeframe, which is acceptable for some optimization problems.

In general, meta-heuristics can be classified into single solution-based meta-heuristic algorithms and population-based meta-heuristic algorithms. In the first case, a single solution is optimized over a series of iterations, e.g., the taboo search (TS) algorithm and the simulated annealing (SA) algorithm.

While in the latter case, a pool of solutions is called a “population,” where the search process starts with a randomly generated initial population and is then optimized over a series of iterations that continue until some stopping criteria are met, e.g., a genetic algorithm (GA) or local search (LS).

A subclass of population-based meta-heuristics that are inspired by the cooperative behavior of species is referred to as “swarm intelligence” algorithms such as Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and Particle Swarm Optimizer (PSO) [1]. This approach consists of simple and homogeneous search agents called particles moving in the search space, which interact with the environment and indirectly communicate with each other.

PSO is one of the popular algorithms inspired by swarm intelligence, which benefits from cooperative behavior to avoid predators or seek food in natural organisms such as fish schooling or bird flocking.

Gray Wolf Optimizer (GWO) [2] is another swarm intelligence-based algorithm that mimics the hunting mechanisms and social hierarchy of gray wolves in nature. GWO has been used to solve various problems, e.g., global optimization problems, electrical and power engineering problems, scheduling problems, power dispatch problems, control engineering problems, and many others [3].

The aim of this paper is to make a comparative study of the improvements proposed in the literature to address the GWO weaknesses.

This paper is organized as follows: The inspiration and mathematical model of GWO are presented in Sect. 2. In Sect. 3, the summarization and classification of the collected literature into four categories are described. The impact of the improvements on the original GWO for each proposed variant in the literature was presented in Sect. 4. Finally, the main findings of the presented study and the possible future work are presented in Sect. 5.

2. Gray wolf Optimizer

This section, the inspiration for the GWO algorithm, and the mathematical model presented in Subsections 2.1 and 2.2 explain the algorithm's procedures.

2.1 GWO inspiration and mathematical model

The Gray Wolf optimizer is a meta-heuristic based on swarm intelligence that was recently developed by Mirjalili et al. [2]. The GWO mimics gray wolf social behavior and focuses on the social hierarchy of gray wolves and the group's hunting mechanism.

The gray wolf social hierarchy consists of four layers of dominance: the alpha wolf, who has the most dominance; the beta wolf, who dominates the pack after the alpha and serves as an advisor to the alpha; the delta wolf, who obeys alpha and beta and dominates the lowest layer; and finally, the omega wolves, who constitute the fourth layer and are the lowest level in the pack.

Gray wolves hunt in packs, with the key phases of the hunting process being tracking, surrounding, and attacking.

The social hierarchy is mathematically modeled by representing the three fittest solutions as alpha, beta, and delta, respectively. The rest of the population is made up of omega wolves.

The gray wolf begins the hunt by encircling the prey; this behavior is mathematically modelled in Eqs. (1 and 2).

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (1)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (2)$$

\vec{D} Represent the distance between the wolf and the prey, $\vec{X}(t+1)$ indicate to the new position of the wolf, \vec{X}_p, \vec{X} indicate the prey position vector and wolf position vector, respectively. t Is the current iteration and \vec{A}, \vec{C} are coefficient vectors and calculated as given in Eqs. (3 and 4).

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (3)$$

$$\vec{C} = 2 \cdot \vec{r}_2 \quad (4)$$

\vec{r}_1, \vec{r}_2 are two random vectors in the range (0, 1), and \vec{a} is a vector that controls the exploration and exploitation and linearly decreases from 2 to 0, representing the magnitude of movement and calculated as given in Eq. (5).

$$\vec{a} = 2 - \left(\frac{2 \times \text{current iteration}}{\text{max iteration}} \right) \quad (5)$$

Since alpha, beta, and delta are the best members of the pack, they have a superior understanding of the prey's location (optima). Therefore, during hunting, these wolves lead the search process, and the omega wolves adjust their positions based on the positions of the fittest three wolves in the pack, as given in Eqs. (6, 7, and 8) [2].

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|, \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|, \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \quad (6)$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha, \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta, \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \quad (7)$$

$$\vec{X}(t + 1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (8)$$

2.2 GWO procedures

The GWO started the search process by randomly initializing the swarm. The best three solutions, namely alpha, beta, and delta, estimate the prey's location. All other wolves update their distances from the prey. At each iteration, the objective function is calculated for each wolf, and the alpha, beta, and delta wolves are updated. The wolf position is then updated towards the leading wolves using the position update equations. The exploration and exploitation are guaranteed through the decreasing values of \vec{a} and \vec{A} parameters with iteration progress. The \vec{C} parameter help the search agents by providing random values between 0 and 2 that assist the exploration even in the last iterations. Finally, when some stopping criteria are met, the algorithm returns the alpha position vector as the best-found solution. **Figure 1** shows the procedures of Basic GWO [2].

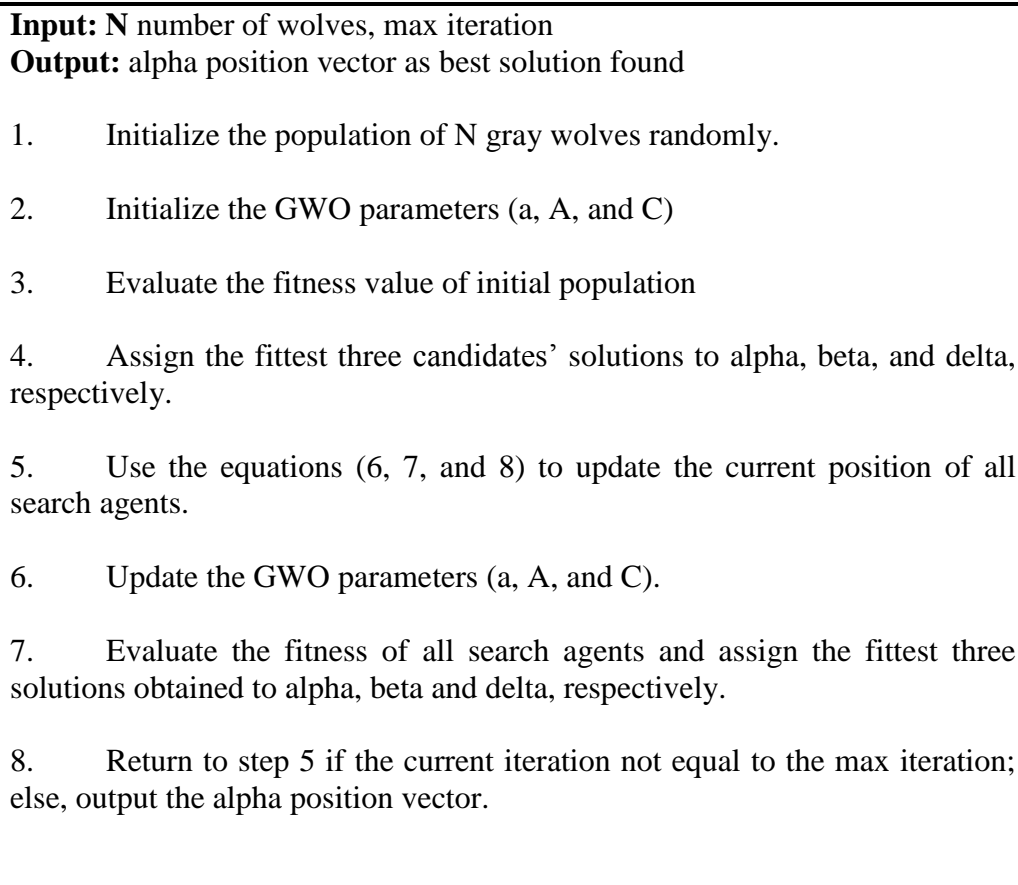


Figure 1: GWO algorithm.3. New variants of GWO

Many variants of the gray wolf optimizer have been proposed and used in various fields. These variants attempted to improve various aspects of GWO by adopting and suggesting various techniques. Faris et al. [3] In 2017, they published a review paper on improved versions and applications of GWO. Therefore, in this study, we follow their efforts, but we focus on the new variants of GWO and its applications in recent years, especially the years from 2018 to 2021, without including the applications of the original GWO.

The reviewed studies were categorized based on the categorization of Faris et al. [3], with some modifications:

- **Modifying conventional algorithm's equations:** such as the position update equation or the control parameter equation
- **Employing new mechanisms:** incorporating additional mechanisms gleaned from other algorithms.
- **Multiple modifications:** the studies in this category focused on more than two aspects and used the modification of conventional algorithm equations as well as mechanisms from other algorithms.
- **Binary or discrete GWO variant.**

The studies were gathered from a variety of well-regarded publishers, such as Elsevier, Springer, IEEE, MDPI, and others. The keywords used in the search using the Google Scholar database are "improved Gray Wolf optimizer" and "enhanced Gray Wolf optimizer."

3.1 Modifying conventional algorithm's equation

Several works of literature suggested a new position update mechanism in an attempt to improve the original algorithm.

In [4], Hosseini et al. use the positions of the former alpha and beta as co-decision makers along with the current alpha, beta, and delta to update the position of the current individual. The effect of the former alpha and beta is linearly faded out using the control parameter α according to Eq. (9).

$$\vec{X}(t+1) = \frac{\vec{X}_1(t) + \vec{X}_2(t) + \vec{X}_3(t) + \left(\frac{\vec{a}(t)}{2}\right) \cdot (\vec{X}_1(t-1) + \vec{X}_2(t-1))}{3 + \vec{a}(t)} \quad (9)$$

Kumar et al. [5] proposed an enhanced version of GWO that is used for feature selection with SVM. The two main enhancements they proposed are using the fitness-sharing technique, where the fitness of a wolf is shared among other wolves, encircling a similar solution. This aids in avoiding convergence to local optima and locating all of an objective function's global optima. The concept of a weighted position update is also introduced to improve the performance of the base GWO.

A different approach was used in [6]. According to Eq. 10, Seyyedabbasi et al. developed two versions of GWO: the first is called the "expanded Gray Wolf Optimizer," and it is an expanded model of the GWO algorithm in which the i 'th agent updates its own position based on $(n-1)$ the previous agent's position in the pack, where n starts at 4 and the first three represent alpha, beta, and delta, respectively. The second algorithm is called incremental GWO, which is based on the incremental model. In this method, each agent updates its own position based on all the agents selected before it, according to Eq. 11.

$$\vec{X}_n(t+1) = \frac{1}{n-1} \sum_{i=1}^{n-1} X_i(t); \quad n = 4, 5, \dots, m \quad (10)$$

$$\vec{X}_n(t+1) = \frac{1}{n-1} \sum_{i=1}^{n-1} X_i(t); \quad n = 2, 3, \dots, m \quad (11)$$

Alejo Reyes et al. [7] suggest a new position update mechanism for GWO to solve the complex optimization problems of supplier selection and order quantity allocation, which is a discrete problem. They claimed that the proposed mechanism increases and improves the explorative properties of the original GWO and maintains its important characteristics so that the algorithm can converge to difficult, high multi-modal optima. Two additional elements

have been included in the position update equation (weighted factors and a displacement vector), as shown in Eq. (12). The weights are assigned in different proportions for alpha, beta, and delta wolves based on the hierarchy. The displacement vector ($\vec{r}_3 \vec{b}$) has been included in order to increase the exploration and prevent the consideration of unfeasible solutions.

$$\vec{X}(t+1) = w_1 \vec{X}_1 + w_2 \vec{X}_2 + w_3 \vec{X}_3 + \vec{r}_3 \vec{b} \quad (12)$$

In [8], the authors proposed two key improvements for GWO to ensure the accuracy of the solution: A more local search is obtained to improve the accuracy of the solution by using Eq. (13) as the control parameter, which produces a global search in 38% of iterations and a local search in 62% of iterations. More population diversity is achieved through a new position update, Eq. (14), which includes omega wolf information as well as a weight factor for this information that is non-linearly increased as iterations progress.

$$a = 2 - 2 \times \log \left(1 + \frac{(\exp - 1)t}{t_{max}} \right) \quad (13)$$

$$X = (1 - \omega(t)) \cdot \frac{X_1 + X_2 + X_3}{3} + \omega(t) \cdot \frac{\sum_4^k X_i}{k - 3} \quad (14)$$

Where $\omega(t) = c_1 \left(\frac{t}{t_{max}} \right)^\theta$, $\theta > 1$

Luo and Kaiping in [9] argue, based on their observations, that the original GWO has great efficacy in solving problems whose global best solutions are situated at the coordinate system's origin; therefore, they propose an enhanced variant of GWO to overcome the aforementioned issue. A new weight-based Eq. (15) to dynamically estimate the location of the prey, where w is the weights of alpha, beta, and delta, respectively, and ϵ is a simulated stochastic error drawn from a Gaussian distribution. New positions update Eq. (16) for the wolves in the pack, guiding each wolf in the pack directly to the estimated location of the prey, where r is a uniformly distributed random number in $[-2, 2]$ that behaves similarly to the 'a' parameter in the original version.

$$x_p^j = \omega_\alpha * x_\alpha^j(t) + \omega_\beta * x_\beta^j(t) + \omega_\delta * x_\delta^j(t) + \epsilon(t) \quad (15)$$

$$x_i^j(t+1) = x_p^j(t) - r * |x_p^j(t) - x_i^j(t)| \quad (16)$$

Khanum et al. [10] proposed two variants of GWO. The first proposed variant called (λ GWO) which modifies the distance equation as formula (17), where λ new parameter and used to properly scale the distance and maintain the wolves in a better circle, $\vec{Y}_p(g)$, $\vec{Y}(g)$ the position vector of prey and the current wolf. And the position equation modified as formula (18). All three equations, the encircling equations of prey, the position update equation, and the hunting equation were modified based on these modifications. The second proposed variant keeps all the procedures of conventional GWO and uses Minkowski's average in the position update equation instead of the arithmetic mean followed in the original algorithm.

$$\vec{D} = \frac{1}{\lambda} |\vec{C} \vec{Y}_p(g) + \vec{Y}(g)| \quad (17)$$

$$\vec{Y}(g+1) = \frac{1}{\lambda} \{ \vec{Y}_p(g) + \vec{A} \vec{D} \} \quad (18)$$

A different approach is proposed by Long and Wen et al. [11] to improve the performance of GWO, with two enhancements suggested: new positions in the update equation and a

nonlinear control parameter. Inspired by PSO and to benefit from another individual guiding the search process, a new position update mechanism is introduced, where a random individual selected from the population participates with alpha, beta, and delta wolves in order to enhance the global search. As for the control parameter, a nonlinear increasing strategy is used as a modification for the original control parameter in the traditional GWO. It is apparent that the alpha wolf is the best individual with significant dominance privileges and, therefore, has a great impact on the search process. In order to enhance the convergence speed and accuracy of the GWO and based on the aforementioned fact, a new approach is suggested in [12], where the population's evolving process is governed by alpha's update direction (the best individual). Therefore, in alpha-guided GWO (AgGWO), some dimensions of alpha will be changed after the update. If the new alpha is better than the old one, it may be assumed that these dimensions are moving toward a better place; this is called the alpha updating direction (AUD), and this fact is utilized in this approach to update the position of the individuals. Also, the dominance right of Alpha is highlighted through fixed weight factors assigned to each of the fittest three wolves in AgGWO.

Hu, Pin, et al. discovered in [13] that the alpha updating direction (AUD) updates all individuals if the fitness of the current alpha is better than that of the previous one, which is not conducive to diversity and is more likely to cause stagnation problems; thus, they proposed an improved alpha-guided GWO version. In this approach, if the current alpha is better than the previous, then only the fittest three individuals are updated by the alpha update direction, while other individuals are updated using the fixed weight position-update equation in AgGWO. If the current alpha is not better than the previous, then all individuals, including the fittest three, will update using the same position-update equation. The alpha wolf generates two mutant wolves at each iteration by using two mutation operators to replace the two worst individuals in order to overcome the stagnation problem and keep the diversity.

Li and Si-Yu et al. [14] suggested an improved gray wolf optimizer with two main improvements: a new nonlinear control parameter and a new weight-based position-update mechanism. A nonlinear control parameter was adopted for a better balance between exploration and exploitation. The weights in the position-update equation are based on the differences between the fitness values of the leader's wolves. According to Eq. (19), Q represents the fitness value, which is calculated according to Eq. (20), and ψ represents the objective function.

$$x = x_{\alpha} \cdot \left[\frac{Q(\alpha)}{(Q(\alpha) + Q(\beta) + Q(\delta))} \right] + x_{\beta} \cdot \left[\frac{Q(\beta)}{(Q(\alpha) + Q(\beta) + Q(\delta))} \right] + x_{\delta} \cdot \left[\frac{Q(\delta)}{(Q(\alpha) + Q(\beta) + Q(\delta))} \right] \quad (19)$$

$$Q(\alpha) = 1/\psi(x_{\alpha}(t)), \quad Q(\beta) = 1/\psi(x_{\beta}(t)), \quad Q(\delta) = 1/\psi(x_{\delta}(t)) \quad (20)$$

Rodríguez et al. [15] proposed three weight-based position update mechanisms, one based on fitness and two based on fuzzy systems. The first mechanism is based on fitness weights; fitness values are used to generate dynamic weights to adjust the contribution of each leader wolf. The second and third mechanisms are both fuzzy based, and exploit the abilities of fuzzy logic for adjusting the dynamic parameters. The second mechanism assumes that at the beginning all wolf leaders guide the search, and at the end of the search process, the alpha

wolf guides the search. The third mechanism is the opposite of the second, where the alpha wolf guides the search, and at the final phase, all the wolves' leaders are involved in guiding the search process.

3.2 Employing new mechanism

Another way to improve GWO efficiency is to incorporate some mechanisms from cuckoo search, such as crossover, mutation, or levy-flight. Many studies have attempted to improve GWO from this perspective.

In order to sustain variety while improving the balance between local and global search, Nadimi et al. [16] employ a new movement strategy known as the dimension learning-based hunting (DLH) search method, which employs a novel way to build a neighborhood for each wolf in which neighborhood information can be shared between wolves.

Pan and Jiawen et al. [17] proposed a modified version of the GWO that combined the adaptive gray wolf optimization (AGWO) and chaotic gray wolf optimization (CGWO) algorithms, which aims to achieve a higher convergence speed in the GWO. The chaotic algorithm provides an initial population with a uniform distribution and keeps the population's diversity. To keep the balance between exploitation and exploration abilities, a new nonlinear control parameter is proposed as illustrated in Eq. (21).

$$a = 2 - 2 \times \left[\frac{1}{e - 1} \right] \times \left[\exp\left(\frac{t}{m}\right) - 1 \right] \quad (21)$$

Liu et al. [18] integrated GWO with the differential evolution (DE) algorithm and the OTSU algorithm for image recognition problems. In the Gray Wolf optimizer, the proposed algorithm aimed to solve the problems of poor stability and easily falling into the local optimal solution.

Guo et al. [19] used tracking mode and seeking mode to make three variants of GWO, the first based on tracking, the second based on seeking, and the third based on both. The first strategy is based on tracking mode (TGWO), which is used to update the fittest three individuals to increase their global search ability. The second strategy is based on seeking mode (SGWO), where seeking mode is separately added to the gray wolf optimizer. The third strategy is based on both modes (TSGWO). The tracking mode is used to update the position of the alpha wolf, and the seeking mode is used to update the positions of the beta wolf and delta wolf. They claimed this improved convergence accuracy and prevented falling into the local optimum.

In GWO, the fittest three individuals serve as the foundation that leads the search process; therefore, if they are trapped in local optima, the rest of the pack will follow. In an attempt to address the aforementioned issue, Gupta et al. [20] proposed an improved leadership-based GWO where the leader wolves are updated through a Levy-flight search mechanism to avoid getting stuck in local optima, while the omega wolves use the same conventional mechanism with a greedy selection between the previous and present states of the wolves to maintain the strength of the wolf pack and avoid the wolves being diverted from the promising domains of search space. The position updates Eq. (22) for the leaders based on Levy-flight search, where S is the step length, which is drawn from the Levy distribution, and the authors use the Mantegna algorithm for a symmetric Levy distribution. par is a linearly decreasing vector that controls the step length.

$$X'_{ij} = X_{ij} + par \times s \quad (22)$$

Gupta et al. [21] also proposed another variant of GWO that seems similar to the previous variant, but they used the random walk operator instead of the Levy-flight operator.

Differential evolution and elimination mechanisms were utilized by Wang et al. [22] to improve the gray wolf optimizer through a proper balance between exploration and exploitation, accelerate the convergence, and increase the optimization accuracy of the GWO. The differential evolution operations of mutation, crossover, and selection are used to generate the wolves of the next generation, the same as in conventional GWO. The best three individuals are selected as alpha, beta, and delta. In the “survival of the fittest” (SOF) mechanism used to update the wolf pack, after each iteration the individuals are sorted in ascending order, then N wolves with the worst fitness value are eliminated and N wolves are randomly generated, where N is the number of selected wolves in the elimination mechanism and N is a random number within a specific interval identified by the authors.

In [23], an enhanced GWO is proposed for clustering purposes, where the proposed EGWO is parallelized on the MapReduce model to handle the large-scale datasets. The proposed EGWO is hybridized with binomial crossover since the alpha wolf represents the best current position; therefore, a binomial crossover between the other wolves and the alpha wolf is performed to inflate the attack on the prey. Furthermore, some randomness is inducted through Lévy flight in order to empower the exploration capability and reduce the problem of stagnation in local optima. In this study, the Mantegna algorithm is used to produce Lévy flight steps.

Xu et al. [24] benefit from the exploration abilities of Cuckoo Search (CS). The Lévy flight operator is used to update the best three individuals. Their results reveal that the proposed CS-GWO algorithm has greater global search ability and avoids falling into the local optimum.

3.3 Multiple Modifications

Some studies attempted to improve the algorithm through various additions and enhancements by utilizing different techniques and modifications.

Sidea et al. [25] try to solve the optimal scheduling problem for battery energy storage systems (BESS) by introducing the mutationally improved gray wolf optimizer (MIGWO). Multiple modifications are applied to the original GWO that aim to adjust the exploration process in order to reduce the probability of the algorithm stagnating in local minima. Firstly, since the authors were dealing with constrained optimization, two strategies are proposed to generate an initial feasible population. For the problem under consideration, a mutation operator was specifically chosen to improve exploration performance. The mutant wolves will replace only the weakest individuals in the pack by eliminating the least adapted individuals from the population. The social hierarchy of the wolf pack is divided into four layers: the first with one alpha wolf, the second with N beta wolves, the third with N delta wolves, and the fourth with omega wolves. The position update equation is the same as in the conventional GWO; the only addition is that beta and delta wolves are randomly selected from the multiple beta and delta wolves. According to Eq. (23 and 24), the number of beta and delta wolves decreases with each iteration and proceeds until one.

$$N_{\beta} = \max\{\text{round}\{N_{\beta}^{\max} \cdot (1 - l_{act}/l_{max})\}, 1\} \quad (23)$$

$$N_{\delta} = \max\{\text{round}\{N_{\delta}^{\max} \cdot (1 - l_{act}/l_{max})\}, 1\} \quad (24)$$

Yang et al. [26] propose three improvements for multi-objective GWO to enhance the convergence and diversity of solutions. Firstly, in order to enhance the exploration of the initial population, the backward learning strategy is used, which increases the search efficiency. Then a new nonlinear control parameter equation was proposed to increase the exploration, which improves the diversity of solutions. Finally, the Cauchy mutation operator is employed to conduct the enhanced search for the leaders.

Xie et al. [27] proposed an enhanced GWO and used it for designing CNN-LSTM networks for time series analysis. Their variant enhanced the original GWO in four aspects. a nonlinear control parameter, the general behavior of this parameter is capable of yielding larger exploration rates in the first half of iterations as well as smaller exploration rates in the second half of iterations. Chaotic diversification of guiding signals; a sinusoidal chaotic map is used to generate the weight factors within the range of [0.5, 0.9]. Enhanced global position updating rules, Lévy flight, are used to conduct leader enhancement on the dimensions where the determinants are higher than 0.5. The new spiral local exploitation mechanism, local exploitation, and fine-tuning around the alpha wolf were done in the last 20% of the iterations.

Another strategy used by Miao et al. [28] to overcome premature convergence and a tendency to stagnation in local optima in the conventional GWO algorithm is the three major enhancements in the proposed gray wolf optimizer with an enhanced hierarchy (GWO-EH). Self-adaptive weight coefficients based on fitness value are proposed. A new set of position update equations was proposed for the leaders, where the highest-ranking wolf was not allowed to update its position based on the lowest-ranking wolf and some randomness was included to weigh the contributions of each of the leader's wolves. Where delta wolf updates its position with respect to alpha and delta wolves, beta wolf updates its position with respect to alpha wolves, as alpha wolves randomly walk within the search space during iterations by using Levy flight. The last enhancement is used to reposition the three worst omega wolves around the leading wolves.

Feng et al. [29] developed an improved GWO to successfully solve the complicated restricted optimization issue employed in the optimal operation of hydropower systems. In the proposed method, at each iteration, the quasi-oppositional learning is used to obtain the quasi-opposite position for any one individual after all wolves update their positions according to the conventional mechanism in GWO. Then an elite mutation operator is used; first, the two offspring produced by the quasi-oppositional and GWO methods are combined to form a hybrid swarm; second, all the wolves in the hybrid swarm are sorted by fitness value; and finally, the best wolves will enter the next generation while the rest are used in mutation operations. If the size of the swarm is initially doubled, then the first half will be used in the aforementioned operations, and the second half, which is the worst, will be abandoned. As mentioned above, this study deals with constrained optimization; therefore, to modify infeasible solutions, the elastic-ball strategy is proposed.

Sharma et al. [30] employ the opposite-based learning technique to enhance the diversity of GWO. First, individuals are scattered around the search space using random initialization and partial opposition-based initialization; this phase is called opposition-based initialization. Second, at each iteration, opposite-based learning is used to avoid stagnation problems and to jump to new locations, where partial opposites of solutions that satisfy specific conditions are generated and only the best among them are passed on to the next generation; this phase is called opposition-based jumping. Another improvement in this study is the use of an

oscillatory function for the control parameter, where parameter (a) oscillates between [0, 2] for the first 75% of the iteration and then reaches 0.02356 for the remaining iterations.

3.4 Binary or discrete GWO variants

The original GWO is a continuous optimization algorithm; therefore, some studies proposed a discrete variant for discrete optimization or a binary variant to address binary problems such as feature selection, etc.

In the first binary GWO proposed by Emary et al. [31], they proposed two variants of binary GWO and used these two variants to select the optimal feature subset for classification purposes. The first approach is based on the position update equation of the original GWO with binary restrictions, where the wolf's steps toward alpha, beta, and delta are binarized, and then a crossover operator is used to update the current wolf position. The second approach, on the other hand, only updates the wolf position after it has been binarized using the transfer function.

An improved binary GWO is proposed in [32] to solve the dependent task scheduling problem in edge computing; in this binary variant of GWO, a V-shaped transfer function is used to change positions of omega wolves. A new control parameter update mechanism is proposed, which is nonlinearly decreased as the search process progresses. The new mechanism ensures quick convergence in the early phase while decreasing convergence pressure and increasing randomness in the latter phase.

Hu and Pei et al. [33] developed an improved binary GWO by analyzing the range of values for the parameters (A, D) under binary conditions. Based on the results of the analysis, four transfer functions called "V-shaped transfer functions" are introduced, which map the continuous values to a range [0, 1] and then discretize them to 0 and 1 according to the probability. Also, a new updating equation for the control parameter that linearly increases from 0 to 2 has been redefined for the binary version.

A novel discrete GWO that distinguishes between exploration and exploitation through the use of update rules was proposed by Martin et al. [34]. In this GWO, the leader is selected randomly, and then with iteration progress, the likelihood of the major leader being chosen rises at the expense of the other leaders.

4. Analysis and discussion

As in subsection 4.1, an attempt has been made here to clarify the limitations that each study addressed, as well as the improvement's perspective in each study and the study's target field. Subsection 4.2 describes the effects of the improvements proposed in each variant on GWO performance, as well as the improvement techniques and experiments used in each study.

4.1 Summary of GWO improvements

The traditional position updates equation emphasizes repositioning the search agents based on the fittest three agents, resulting in poor exploration abilities. As a result, many improvements, such as [4], [7], and [10], attempt to improve the exploration by modifying the conventional position update equation.

The control parameter also has an important role in the exploration, so the modification of this factor can enhance the exploration, as in [8], [11], and [14]. Other drawbacks for the

conventional position update equation were tackled, such as intensifying the omega wolves in a part of the search space in [6], the search bias in [9], etc., as shown in Table 1. Since the leaders play a crucial role in guiding the search process, changes to the leaders' update mechanism, such as in [19], [20], [21], and [24], have been made to address local optima stagnation or premature convergence, as shown in Table 2, as well as other changes to address other drawbacks.

The diversity of the population plays an important role in tackling the local optima stagnation. The diversity can be increased by an accurate initialization of the initial population or the leader's update mechanism, as well as the solution generation, as shown in Table 3. For the binary optimization, [30] made various changes to the original GWO. In [32], [33] introduced several changes to the transfer function and the control parameter, while [34] proposed discrete optimization using update rules, as shown in Table 4.

Table 1: summary of GWO variants modifying conventional algorithm's equations

Study	The problem encountered	Improvement aspect	Target field
[4]- 2021	poor exploration abilities	Position update equation	Power Dispatch
[5]-2021	premature convergence toward local optima	Position update equation	Feature selection
[6]-2021	convergence of omega wolves to each other	Position update equation	global optimization problems
[7]-2020	poor exploration abilities	Position update equation	supply chain management
[8]-2019	solutions accuracy	1.Position update equation 2. the control parameter	Service composition
[9]-2019	search bias toward the origin of the coordinate system	Position update equation	General optimization
[10]-2019	poor exploration abilities	1.encircling equation 2.hunting equation 3.position update equation	unconstrained optimization
[11]-2018	poor exploration abilities	1.Position update equation 2. the control parameter	high-dimensional numerical optimization
[12]- 2018	1.convergence speed 2.convergence accuracy	Position update equation	General optimization
[13]-2018	convergence to local optima	1.Position update equation	General optimization
[14]-2018	ensure performance quality	1.Position update equation 2. the control parameter	the inversion of geoelectrical data
[15]-2017	study the performance of GWO with new hierarchical operator	Position update equation	General optimization

Table 2: summary of GWO variants employing new mechanism

Study	The problem encountered	Improvement aspect	Target field
[16]-2021	1.poor diversity 2.the imbalance in the exploitation-exploration 3.premature convergence	Neighborhood structure	engineering problems
[17]-2021	1.convergence speed 2. optimization accuracy	1.initial population 2. the control parameter	Parameters identification
[18]-2020	1.poor stability of initial population 2.stagnation in local optima	1. initial population 2.updating the population	image segmentation
[19]-2020	1.Easy to fall in local optima 2. the imbalance in the exploitation-exploration	1. Update leaders 2. position-update equation	function optimization
[20]-2020	premature convergence	Leaders update	global optimization
[21]-2019	premature convergence	Leaders update	general optimization
[22]-2019	1.balance between exploration and exploitation 2.convergence speed 3.optimization accuracy	position update mechanism	general optimization
[23]-2018	1.stagnation at local optima 2.slow convergence rate	position update mechanism	Clustering
[24]-2017	easy to fall in local optimum	Leaders update	high-dimensional optimization

Table 3: summary of GWO variants multiple modifications

Study	The problem encountered	Improvement aspect	Target field
[25]-2021	Stagnation in local minima	1.initial solutions feasibility 2.social hierarchy 3.solution generation	Scheduling
[26]-2020	1.stagnation in local optima 2.poor population diversity 3.multi-objective	1.population initialization 2.leaders updating 3. the control parameter	service composition
[27]-2020	1.stagnation at local optima 2.slow convergence rate	1. the control parameter 2.position update equation 3.leaders updating 4.local exploitation	CNN-LSTM evolving
[28]-2020	1.premature convergence 2.stagnation in local optima	1.position update equation 2.leaders updating 3.the worst solutions handling	Network coverage
[29]-2019	1.convergence rate 2.escaping from local optima 3. constraint handling	1.solutions generation 2.feasibility of solutions	Hydropower operation system
[30]-2017	1.solutions accuracy 2.local optima stagnation	1.population initialization 2.population update 3. the control parameter	numerical optimization

Table 4: summary of GWO variants Binary or discrete GWO variant

Study	The problem encountered	Improvement aspect	Target field
[31]-2016	binary optimization	1.position update mechanism 2. converting the continuous value of search agents to binary	Feature selection
[32]-2019	1.escaping from local optima 2.optimal solution exploitation 3.binary optimization	1.transfer function 2. the control parameter	Scheduling
[33]-2020	1.balance between exploration and exploitation 2.solution quality 3.binary optimization	1.transfer function 2. the control parameter	Feature selection
[34]-2018	Discrete optimization	1.leader updating 2.position-update	General

4.2 Summary of modifications effectiveness

GWO's conventional position-update equation is insufficient in terms of convergence speed, solution quality, and exploration abilities; thus, the use of weight-factors in [5], [7], [12], [13], [14], and [15] resulted in improved convergence speed, solution quality, and exploration abilities to some extent.

A better result in [6], [8], and [11] is due to the use of omega wolves' information in the decision-making process, which improves exploration abilities, but it should be noted that the omega wolves' information must be used in a controlled manner to avoid malicious and unwanted information from participating in the decision-making process, as shown in Table 5. The addition of some randomness to the GWO algorithm by incorporating other algorithm operators such as random walk, Lévy flight, crossover, mutation, etc. reinforced population diversity and exploration ability in [18], [22], and [23]. Others in [20], [21], and [24] employed randomness differently to promote diversity and escape from local optima, where the same operators were used to reinforce the leaders of wolves. Also, different techniques are used for various improvements, as shown in Table 6.

The utilization of different techniques in [25], [26], [27], [28], [29], and [30] to improve different aspects of GWO, such as the diversity of populations, the exploration-exploitation control parameter, leader updates, etc., resulted in better overall performance, as shown in Table 7.

In binary variants, the use of a transfer function is the most frequent technique. In [32], [33], and the second approach in [31], many transfer functions were utilized and suggested, as shown in Table 8.

In the studied literature, the control parameter has undergone several modifications, the majority of which were non-linearly decreasing quantities, which also produced superior results. However, because the control parameter has a great impact on the search space, the equation used to compute it must offer an appropriate balance between exploration and exploitation.

Table 5: GWO results comparison modify the conventional algorithm's equations

Study	Techniques used	Target problem	Dataset used	Objective function	Results
[4]-2021	Modified Position-update equation	Power Dispatch	IEEE 33-bus distribution grid	Minimizing active power loss and costs of RPC devices and DG's reactive power output.	—
[5]-2021	1.Modified Position-update equation 2.fitness sharing	Features selection	WDBC database from UCI Repository.	The classification accuracy of the SVM classifier.	98.24 %
[6]-2021	Modified Position-update equation	global optimization problems	CEC 2014 benchmark functions	Minimizing functions	Success rate: Ex-GWO 39% I-GWO 52%
[7]-2020	Modified Position-update equation	Supplier selection and order quantity allocation	A representative problem from another study	maximize the total profit	—
[8]-2019	1.Modified Position-update 2. the control parameter equations	Energy-Aware Service Composition in Cloud Manufacturing	three schemes are designed by authors for experiments	reducing energy consumption	—
[9]-2019	1.Modified Position-update equation 2.new equation to estimate prey location	Engineering design problem	1.Pressure vessel design 2. Tension/Compression string design	1. minimizing the total production cost of a cylindrical vessel 2. minimizing the weight of a tension/compression	Avg. values 1- 7021.126 2- 0.009872
[10]-2019	1.Modified update equations, 2. Minkowski's average	unconstrained optimization	benchmark functions used in original GWO	Minimizing functions	—
[11]-2018	1. Modified Position-update 2. the control parameter equation	Engineering design problem	welded beam design problem	cost minimizing	Mean value: 2.3817
[12]-2018	1.Modified Position-update equation 2.alpha update direction	Engineering design problem	two stage operational amplifier design	maximize gain parameter	0.1100
[13]-2018	1.Modified Position-update equation	Engineering design problem	two stage operational amplifier design	maximize gain parameter	0.5389

[14]-2018	2.alpha update direction 3.two mutation operator				
	1. Modified Position-update equation 2. the control parameter equation	the inversion of geoelectrical data	The MT survey dataset acquired at the Inner Mongolia	Calculating magneto telluric (MT) DC resistivity and induced polarization (IP)	—
[15]-2017	Modified Position-update equation using fitness-based, fuzzy-based weights factors	Numerical optimization	benchmark functions used in original GWO	Minimizing functions	—

Table 6: GWO results comparison employing new mechanism

Study	Techniques used	Target problem	Dataset used	Objective function	Results
[16]-2021	dimension learning-based hunting (DLH) search strategy	Engineering design problem	1-Pressure vessel design problem 2- Welded beam design problem	1-minimize the cost of material, forming and welding 2- minimum fabrication cost	1-5888.3400 2- 1.724853
[17]-2021	1- The chaotic GWO with logistic mapping 2- nonlinear decreasing equation	photovoltaic cells parameters identification	experimental dataset from other literature	Root-mean-square error For three scenarios: 1-single DM 2-double DM 3-three DM	1-0.002877 2- 0.003263 3- 0.002838
[18]-2020	1-Differential evolution 2-the OTSU algorithm 3- Tsallis entropy	Image Recognition	Local data	threshold segmentation 1- (GCE) 2-(RI) 3- (VI)	1-0.0518 2-0.9493 3-0.3095
[19]-2020	1.tracking mode 2.seeking mode	Engineering design problem	Pressure vessel design problem	reduce the cost on the premise of safety	TGWO 5909.2125 SGWO 6411.4448 TSGWO 8706.137
[20]-2020	1.Lévy flight 2. greedy selection	Engineering design problem	1.Design of three bar trusses 2. Design of speed reducer	1. minimizing volume of the truss structure 2. minimizing weight of speed reducer	1. 263.8969 2. 2996.3580
[21]-2019	1.random walk 2. greedy selection	Engineering design problem	1. Gear train design 2.tension/	1.optimal number of teeth for four gears of a train 2. minimizing the	1- 2.7009x10 ⁻¹² 2-

			compression string design problem	weight of a tension/compression spring	0.012674
[22]-2019	1.differential evolution 2. elimination mechanism	Benchmark functions	From other literature	Minimization	—
[23]-2018	1. binomial crossover 2.Lévy flight 3. MapReduce architecture	Clustering	seven datasets from UCI: 1.Iris 2.Seeds 3.Glass 4.Cancer 5.Balance 6.Haberman 7.Wine	sum of squared Euclidean distance	Mean: 1-99.55645 2-311.79804 3-242.68894 4-2964.3869 5-1424.2047 6-2637.3490 7-16292.350
[24]-2017	Lévy flight	Features selection	NSL-KDD datasets	classification accuracy	83.54%

Table 7: GWO results comparison multiple modifications

Study	Techniques used	Target problem	Dataset used	Objective function	Results
[25]-2021	1. two strategies for feasible solutions 2. elimination of the worst individuals mechanism 3.a mutation operator 4. Modified social hierarchy	optimal scheduling of energy storage	the modified IEEE 33-bus test network	minimizing the active power losses	—
[26]-2020	1.backward learning strategy 2.Cauchy's mutation operator 3.nonlinear equation	service composition	randomly generated data	maximal QoS and minimal energy consumption	—
[27]-2020	1. a chaotic weight allocation mechanism 2. nonlinear equation for the control parameter 3. Lévy flight 4. spiral local exploitation scheme	Evolving CNN-LSTM for prediction and classification	HAR dataset from UCI	minimizing error rate	average accuracy 0.923
[28]-2020	1.modified position-update equation 2.new leaders update equation 3.Lévy flight 4.reposition mechanism	WSN coverage problem	three Scenarios of WSN coverage proposed by author	coverage rate	Mean values Case1: 0.9781 Case2: 0.9126 Case3: 0.836

[29]-2019	1.quasi-oppositional learning mechanism 2.new elite mutation operator 3.elastic-ball strategy to adjust infeasible Agents	scheduling of Hydropower Operation	Dataset from "Wu" hydropower system in southwest China	minimizing the standard deviation of the residual load series	Avg. values Case1:36477 Case2:34759 Case3:32711 Case4:31721
[30]-2017	1.opposite-based learning initialization 2. opposite-based learning jumping 3.oscillatory function	Numerical Optimization	benchmark function	Minimizing	—

Table 8: GWO results comparison Binary or discrete variants

Study	Techniques used	Target problem	Dataset used	Objective function	Results
[31]-2016	1. new binary-based position-update equations 2. sigmoidal transfer function	Feature selection	Eighteen datasets from UCI	minimizing error rate and selected features ration	Mean values bGWO-1 First dataset: 0.030 Second dataset: 0.037 bGWO2 First dataset: 0.027 Second dataset: 0.031
[32]-2019	1. V-shaped transfer function 2. nonlinearly decrease equation for the control parameter	Scheduling problem	scenario proposed by authors for dependent task scheduling in edge computing	Minimizing Makespan	Avg. value for 500 subtasks: 2500
[33]-2020	1.new four V-shaped transfer function 2.linearly increasing equation for the control parameter	Feature selection	twelve datasets from UCI	classification error and the number of the features subset	—
[34]-2018	1. leaders selection mechanism 2. update rules	Numerical optimization	Benchmark function	Minimizing	—

5. Conclusions

The Gray Wolf Optimizer is a recent optimization algorithm that belongs to the family of swarm intelligence algorithms. Due to its simplicity, it has been used in a variety of optimization problems.

However, the proposal of new variants of GWO in recent years shows that there are many more ways to make improvements to GWO in order to improve the algorithm’s performance.

Many studies have addressed the limitations of the GWO algorithm. For example, increasing the diversity of populations, achieving a better balance between exploration and exploitation, controlling search direction, or proposing a variant that works on optimization problems not addressed by the original algorithm, such as a binary optimization, discrete optimization, or multi-objective optimization problem.

This study provides an overview of recent GWO variants that aim to improve some of the algorithm's original limitations by providing an overview of the GWO algorithm. Also, we have explained the basic concepts of GWO, along with a variety of advances in the GWO algorithm.

From the presented literature survey on GWO, it can be seen that many studies have been done on the GWO algorithm, and further improvements in GWO performance are possible in future research.

such as.

- Solving constrained optimization requires further investigations with the GWO algorithm.
- The GWO algorithm mainly works in the continuous search space; hence, the area of discrete optimization using this algorithm has little research activity.
- Further research on some aspects of the algorithm, such as the structure of the social hierarchy and the neighborhood structure, could lead to a better performance of the GWO algorithm.
- Involving other search agents in the position update mechanism could benefit the search process and lead to more accurate solutions.
- GWO has great intensification ability but poor diversification ability, so hybridization of GWO with current algorithms that have great exploration ability, such as the genetic algorithm, is an option to achieve diversification.

Finally, taking into account the problem variables, the Gray Wolf optimizer could be a viable candidate for solving a variety of NP-hard problems.

References

- [1] E.-G. Talbi, *Metaheuristics: from design to implementation*. John Wiley & Sons, 2009.
- [2] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46-61, 2014.
- [3] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: a review of recent variants and applications," *Neural computing and applications*, vol. 30, no. 2, pp. 413-435, 2018.
- [4] S. Hosseini-Hemati, G. H. Sheisi, and S. Karimi, "Allocation-based optimal reactive power dispatch considering polynomial load model using improved gray wolf optimizer," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 45, no. 3, pp. 921-944, 2021.
- [5] S. Kumar and M. Singh, "Breast cancer detection based on feature selection using enhanced grey Wolf optimizer and support vector machine algorithms," *Vietnam Journal of Computer Science*, vol. 8, no. 02, pp. 177-197, 2021.
- [6] A. Seyyedabbasi and F. 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.
- [7] A. Alejo-Reyes, E. Cuevas, A. Rodríguez, A. Mendoza, and E. Olivares-Benitez, "An improved grey wolf optimizer for a supplier selection and order quantity allocation problem," *Mathematics*, vol. 8, no. 9, p. 1457, 2020.
- [8] Y. Yang, B. Yang, S. Wang, W. Liu, and T. Jin, "An improved grey wolf optimizer algorithm for energy-aware service composition in cloud manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 105, no. 7, pp. 3079-3091, 2019.

- [9] K. Luo, "Enhanced grey wolf optimizer with a model for dynamically estimating the location of the prey," *Applied Soft Computing*, vol. 77, pp. 225-235, 2019.
- [10] R. A. Khanum, M. A. Jan, A. Aldegheishem, A. Mehmood, N. Alrajeh, and A. Khanan, "Two new improved variants of grey wolf optimizer for unconstrained optimization," *IEEE Access*, vol. 8, pp. 30805-30825, 2019.
- [11] W. Long, J. Jiao, X. Liang, and M. Tang, "An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization," *Engineering Applications of Artificial Intelligence*, vol. 68, pp. 63-80, 2018.
- [12] P. Hu, S. Chen, H. Huang, Z. Xiao, and S. Huang, "Alpha guided grey wolf optimizer and its application in two stage operational amplifier design," in *2018 13th World Congress on Intelligent Control and Automation (WCICA)*, IEEE, pp. 560-565, 2018.
- [13] P. Hu, S. Chen, H. Huang, G. Zhang, and L. Liu, "Improved alpha-guided Grey wolf optimizer," *IEEE Access*, vol. 7, pp. 5421-5437, 2018.
- [14] S.-Y. Li, S.-M. Wang, P.-F. Wang, X.-L. Su, X.-S. Zhang, and Z.-H. Dong, "An improved grey wolf optimizer algorithm for the inversion of geoelectrical data," *Acta Geophysica*, vol. 66, no. 4, pp. 607-621, 2018.
- [15] L. Rodríguez et al., "A fuzzy hierarchical operator in the grey wolf optimizer algorithm," *Applied Soft Computing*, vol. 57, pp. 315-328, 2017.
- [16] M. H. Nadimi-Shahraki, S. Taghian, and S. Mirjalili, "An improved grey wolf optimizer for solving engineering problems," *Expert Systems with Applications*, vol. 166, p. 113917, 2021.
- [17] J. Pan, Y. Gao, Q. Qian, Y. Feng, Y. Fu, and F. Sardari, "Parameters identification of photovoltaic cells using improved version of the chaotic grey wolf optimizer," *Optik*, vol. 242, p. 167150, 2021.
- [18] Y. Liu, J. Sun, H. Yu, Y. Wang, and X. Zhou, "An improved grey wolf optimizer based on differential evolution and OTSU algorithm," *Applied Sciences*, vol. 10, no. 18, p. 6343, 2020.
- [19] M. Guo, J.-S. Wang, L. Zhu, S.-S. Guo, and W. Xie, "An improved grey wolf optimizer based on tracking and seeking modes to solve function optimization problems," *IEEE Access*, vol. 8, pp. 69861-69893, 2020.
- [20] S. Gupta and K. Deep, "Enhanced leadership-inspired grey wolf optimizer for global optimization problems," *Engineering with Computers*, vol. 36, no. 4, pp. 1777-1800, 2020.
- [21] S. Gupta and K. Deep, "A novel random walk grey wolf optimizer," *Swarm and evolutionary computation*, vol. 44, pp. 101-112, 2019.
- [22] J.-S. Wang and S.-X. Li, "An improved grey wolf optimizer based on differential evolution and elimination mechanism," *Scientific reports*, vol. 9, no. 1, pp. 1-21, 2019.
- [23] A. K. Tripathi, K. Sharma, and M. Bala, "A novel clustering method using enhanced grey wolf optimizer and mapreduce," *Big data research*, vol. 14, pp. 93-100, 2018.
- [24] H. Xu, X. Liu, and J. Su, "An improved grey wolf optimizer algorithm integrated with cuckoo search," in *2017 9th IEEE international conference on intelligent data acquisition and advanced computing systems: technology and applications (IDAACS)*, IEEE, vol. 1, pp. 490-493, 2017.
- [25] D. O. Sidea, I. I. Picioroaga, and C. Bulac, "Optimal battery energy storage system scheduling based on mutation-improved grey wolf optimizer using gpu-accelerated load flow in active distribution networks," *IEEE Access*, vol. 9, pp. 13922-13937, 2021.
- [26] Y. Yang, B. Yang, S. Wang, T. Jin, and S. Li, "An enhanced multi-objective grey wolf optimizer for service composition in cloud manufacturing," *Applied Soft Computing*, vol. 87, p. 106003, 2020.
- [27] H. Xie, L. Zhang, and C. P. Lim, "Evolving CNN-LSTM models for time series prediction using enhanced grey wolf optimizer," *IEEE Access*, vol. 8, pp. 161519-161541, 2020.
- [28] Z. Miao, X. Yuan, F. Zhou, X. Qiu, Y. Song, and K. Chen, "Grey wolf optimizer with an enhanced hierarchy and its application to the wireless sensor network coverage optimization problem," *Applied Soft Computing*, vol. 96, p. 106602, 2020.
- [29] Z.-K. Feng et al., "Optimal operation of hydropower system by improved grey wolf optimizer based on elite mutation and quasi-oppositional learning," *IEEE Access*, vol. 7, pp. 155513-155529, 2019.

- [30] S. Sharma, R. Salgotra, and U. Singh, "An enhanced grey wolf optimizer for numerical optimization," in *2017 international conference on innovations in information, embedded and communication systems (ICIIECS)*, IEEE, pp. 1-6, 2017.
- [31] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371-381, 2016.
- [32] K. Jiang, H. Ni, P. Sun, and R. Han, "An improved binary grey wolf optimizer for dependent task scheduling in edge computing," in *2019 21st International Conference on Advanced Communication Technology (ICACT)*, IEEE, pp. 182-186, 2019.
- [33] P. Hu, J.-S. Pan, and S.-C. Chu, "Improved binary grey wolf optimizer and its application for feature selection," *Knowledge-Based Systems*, vol. 195, p. 105746, 2020.
- [34] B. Martin, J. Marot, and S. Bourennane, "Improved discrete grey wolf optimizer," in *2018 26th European Signal Processing Conference (EUSIPCO)*, IEEE, pp. 494-498, 2018.