



Grey wolf optimizer: a review of recent variants and applications

Hossam Faris¹ · Ibrahim Aljarah¹ · Mohammed Azmi Al-Betar² · Seyedali Mirjalili³ 

Received: 19 June 2017 / Accepted: 13 November 2017
© The Natural Computing Applications Forum 2017

Abstract

Grey wolf optimizer (GWO) is one of recent metaheuristics swarm intelligence methods. It has been widely tailored for a wide variety of optimization problems due to its impressive characteristics over other swarm intelligence methods: it has very few parameters, and no derivation information is required in the initial search. Also it is simple, easy to use, flexible, scalable, and has a special capability to strike the right balance between the exploration and exploitation during the search which leads to favourable convergence. Therefore, the GWO has recently gained a very big research interest with tremendous audiences from several domains in a very short time. Thus, in this review paper, several research publications using GWO have been overviewed and summarized. Initially, an introductory information about GWO is provided which illustrates the natural foundation context and its related optimization conceptual framework. The main operations of GWO are procedurally discussed, and the theoretical foundation is described. Furthermore, the recent versions of GWO are discussed in detail which are categorized into modified, hybridized and paralleled versions. The main applications of GWO are also thoroughly described. The applications belong to the domains of global optimization, power engineering, bioinformatics, environmental applications, machine learning, networking and image processing, etc. The open source software of GWO is also provided. The review paper is ended by providing a summary conclusion of the main foundation of GWO and suggests several possible future directions that can be further investigated.

Keywords Optimization · Metaheuristics · GWO

1 Introduction

Computational intelligence (CI) [24] is a sub-field of the artificial intelligence (AI), which consists of a variety of mechanisms to solve complex problems in different environments. CI concerns with solving mechanism itself to produce acceptable solutions, while AI focuses on the outcome of the mechanism that produces optimal solutions. CI includes many forms such as artificial neural networks (ANN) [40], evolutionary computation (EC) [3] and swarm intelligence (SI) [23].

EC and related evolutionary algorithms (EAs) are considered the fastest growing algorithms that employ CI to solve optimization problems. EC is a category of algorithms inspired by biological evolution and natural selection techniques. Most of the evolutionary computation algorithms called population-based algorithms such as the algorithm start with a population of random solutions. Then, the population is evolved using a set of evolution operators such as mutation and recombination operators to enhance the quality of the population and achieve better

✉ Seyedali Mirjalili
seyedali.mirjalili@griffithuni.edu.au

Hossam Faris
hossam.faris@ju.edu.jo

Ibrahim Aljarah
i.aljarah@ju.edu.jo

Mohammed Azmi Al-Betar
mohbetar@bau.edu.jo

¹ Business Information Technology Department, King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan

² Department of Information Technology, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan

³ Institute for Integrated and Intelligent Systems, Griffith University, Nathan, Brisbane, QLD 4111, Australia

solutions at the end of the algorithm. The EAs include genetic algorithm (GA) [11], evolutionary strategies (ES) [37], genetic programming (GP) [59], population-based incremental learning (PBIL) [38], biogeography-based optimizer (BBO) [116] and differential evolution (DE) [120].

Swarm intelligence (SI) [23] is another powerful form of the CI used to solve the optimization problems. SI algorithms simulate and imitate the natural swarms or communities or systems such as fish schools, bird swarms, bacterial growth, insects colonies and animal herds. Most of the SI algorithms concentrate on the behavior of swarm's members and their life style besides the interactions, and relations between the swarm's members to locate the food sources. SI algorithms include many algorithms such as ant colony optimization (ACO) [13], particle swarm optimization (PSO) [15], cuckoo search (CS) [138], krill herd optimization (KH) [29], firefly algorithm (FA) [137], artificial bee colony (ABC) [50], multi-verse optimizer (MVO) [82], ant lion optimizer (ALO) [76], sine cosine algorithm (SCA) [80], dragonfly algorithm (DA) [79], whale optimization algorithm (WOA) [81], moth-flame optimization (MFO) [78], grey wolf optimizer (GWO) [83] and many others.

GWO is one of the recently proposed swarm intelligence-based algorithms, which is developed by Mirjalili et al. [83] in 2014. The GWO algorithm is inspired by grey wolves in nature that searching for the optimal way for hunting preys. GWO algorithm applies the same mechanism in nature, where it follows the pack hierarchy for organizing the different roles in the wolves pack. In GWO, pack's members are divided into four groups based on the type of the wolf's role that help in advancing the hunting process. The four groups are alpha, beta, delta and omega, where the Alpha represents the best solution found for hunting so far. The division of population to four groups is done in the original GWO paper to comply with the dominance hierarchy of grey wolves in nature. The inventors of this algorithm conducted an extensive experiment and observed that considering four groups results in the best average performance on benchmark problems and a set of low-dimensional real-world case studies. However, considering more or less groups can be investigated as a future work when solving large-scale challenging problems.

The GWO search process like other SI-based algorithms starts with creating random population of grey wolves. After that, the four groups of wolves and their positions are formed and the distances to the target prey are measured. Each wolf represents a candidate solution and is updated through the searching process. Furthermore, GWO applies powerful operations controlled by two parameters to

maintain the exploration and exploitation to avoid the local optima stagnation.

Despite the similarity in the way that GWO estimates the global optimum compared to other population-based algorithms, the mathematical model of this algorithm is novel. It allows relocating a solution around another in an n -dimensional search space to simulate chasing and encircling preys by grey wolves in nature.

GWO has one vector of position, so it requires less memory compared to PSO with position and velocity vectors. Also, GWO saves only three best solutions, while PSO saves one best solution for each particle as well as the best solution obtained so far by all particles. The mathematical equations of PSO and GWO are different.

GWO is considered as one of the most growing SI algorithms. The prosperity of GWO algorithm motivates other researchers to apply the algorithm for solving different types of optimization problems. The GWO is used to solve different problems such as the global optimization problems, electric and power engineering problems, scheduling problems, power dispatch problems, control engineering problems, robotics and path planning problems, environmental planning problems and many others.

The main objective of this review is to make a comprehensive study of all aspects of GWO algorithm, and how the researchers are motivated to apply it in different applications. In addition, this review will highlight the strengths of algorithm and the improvements suggested in the literature to overcome the algorithm weaknesses. Furthermore, the review will refer to all of the previous research that discussed the GWO by referring to the various well-regarded publishers such as Elsevier, Springer, IEEE, Taylor & Francis, Inderscience, Hindawi and others. Figure 1 shows the number of publications, which are distributed based on the publisher of the GWO-related articles. Figure 2 shows the distribution of these publications based on the type of application.

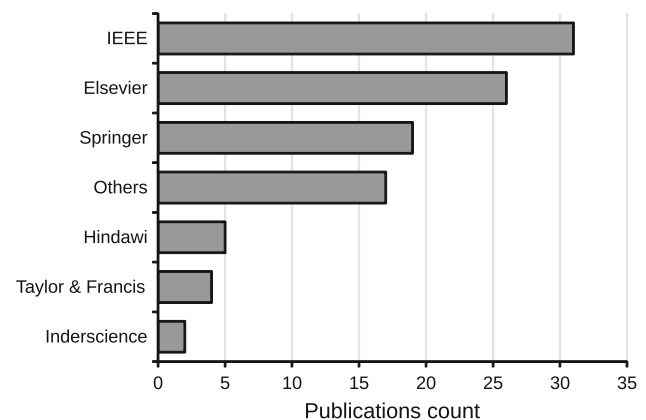


Fig. 1 Number of publications of GWO per publisher

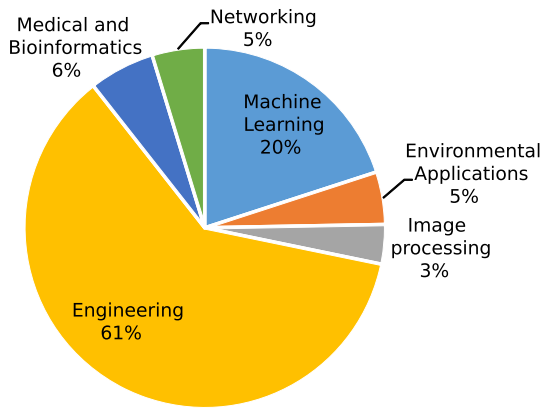


Fig. 2 GWO applications

This review discusses GWO algorithm based on three classifications:

- *Theoretical aspects of GWO* which includes the GWO modifications, hybridized versions of GWO, parallel versions of GWO and multi-objective versions of GWO. Figure 3 shows the classification of the theoretical aspects of GWO.
- *Applications of GWO* which includes engineering applications, networking applications, environmental applications, medical and bioinformatics applications, machine learning applications and image processing application. Figure 2 shows the number of publications, which are distributed based on the type of the application.
- Open source softwares, libraries, frameworks and toolboxes of GWO.

This paper is organized as follows: The preliminary background concerning the GWO and its foundation is presented in Sect. 2. In Sect. 3, the theoretical aspects of GWO and improvement details are described. In Sect. 4,

applications of GWO are described and highlighted. GWO open source libraries and frameworks are presented in Sect. 5. Finally, in Sect. 6, the future works and possible research directions of GWO are given.

2 Grey wolf optimizer

This section presents the GWO algorithm and describes its main components. It also includes discussions on the exploration, exploitation and convergence of this algorithm.

2.1 Inspiration of GWO

GWO is a swarm intelligence technique. The inspiration of the GWO algorithm is the social intelligence of grey wolf packs in leadership and hunting. In each pack of grey wolves, there is a common social hierarchy that dictates power and domination (see Fig. 4).

The most powerful wolf is alpha, which leads the entire pack in hunting, migration and feeding. When the alpha wolf is not in the pack, ill, or dead, the strongest wolf from the beta wolves takes the lead of the pack. The power and domination of delta and omega are less than alpha and beta as can be seen in Fig. 4. This social intelligence is the main inspiration of the GWO algorithm.

Another inspiration is the hunting approach of grey wolves. When hunting a prey, grey wolves follow a set of efficient steps: chasing, encircling, harassing and attacking. This allows them to hunt big preys.

2.2 Mathematical models of GWO

This section briefly presents the mathematical model of the GWO algorithm.

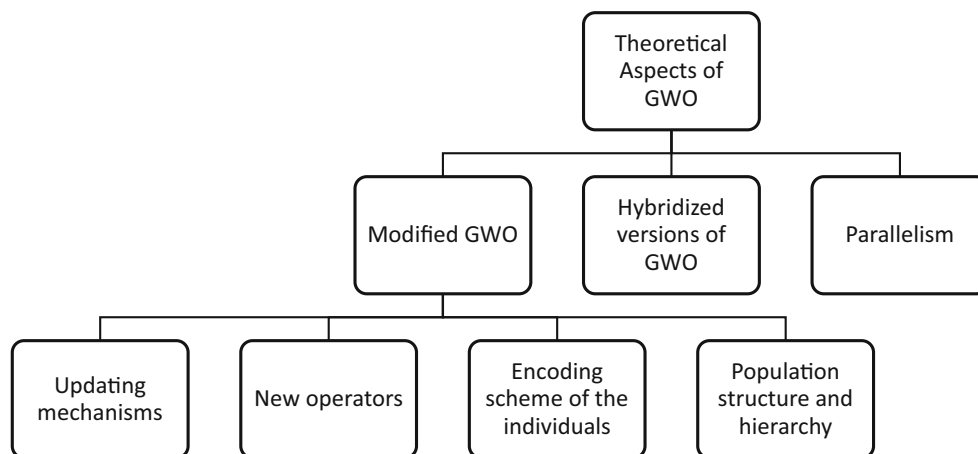


Fig. 3 Classification of the theoretical aspects of GWO algorithm

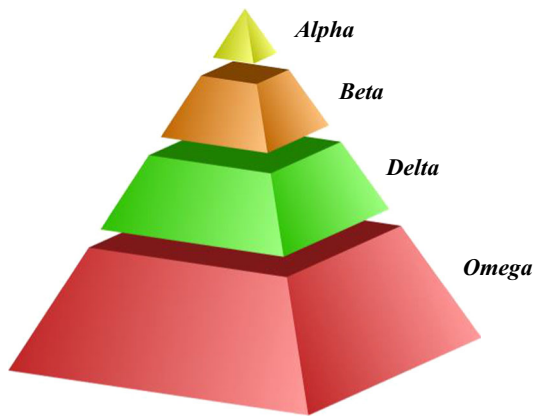


Fig. 4 Social hierarchy of grey wolves

2.2.1 Encircling prey

As mentioning above the first step of hunting is to chase and encircle. To mathematically model this, GWO considers two points in a n -dimensional space and updates the location of one of them based on that of another. The following equation has been proposed to simulate this:

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

where $\mathbf{X}(t+1)$ is the next location of the wolf, $\mathbf{X}(t)$ is current location, \mathbf{A} is a coefficient matrix and \mathbf{D} is a vector that depends on the location of the prey (\mathbf{X}_p) and is calculated as follows:

$$\mathbf{D} = |\mathbf{C} \cdot \mathbf{X}_p(t) - \mathbf{X}(t)| \quad (2)$$

where

$$\mathbf{C} = 2 \cdot \mathbf{r}_2$$

Note that \mathbf{r}_2 is a randomly generated vector from the interval $[0,1]$. With these two equations, a solution is able to relocate around another solution. Note that the equations use vectors, so this is applied to any number of dimension. An example of possible positions of a grey wolf with respect to a prey is shown in Fig. 5.

Note that the random components in the above equations simulate different step sizes and movement speeds of grey wolves. The equations to define their values are as follows:

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

where \mathbf{a} is a vector where its values are linearly decreased from 2 to 0 during the course of run. \mathbf{r}_1 is a randomly generated vector from the interval $[0,1]$. The equation to update the parameter a is as follows:

$$a = 2 - t \left(\frac{2}{T} \right) \quad (4)$$

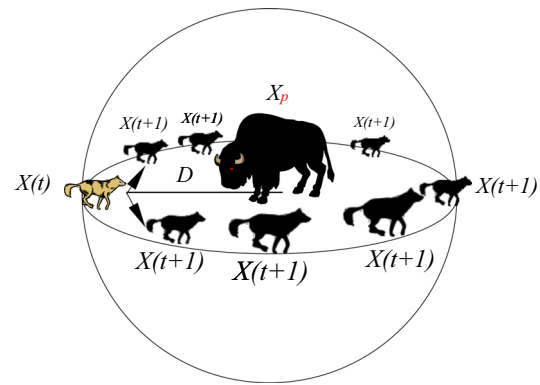


Fig. 5 How the mathematical equations allow position updating around a pivot point. Eq. 1 mathematically models the position updating of a grey wolf ($X(t)$) around a prey (X_p). Depending on the distance between the wolf and the prey (D), a wolf can be relocated in a circle (in a 2D space), sphere (in a 3D space), or a hypersphere (in an N -D space) around the prey (X_p) using Eq. 1

where t shows the current iteration and T is the maximum number of iterations.

2.2.2 Hunt

With the equations presented above, a wolf can relocate to any points in a hypersphere around the prey. However, this is not enough to simulate the social intelligence of grey wolves. It was discussed above that social hierarchy plays a key role in hunt and the survival of a packs. To simulate social hierarchy, three best solutions are considered to be alpha, beta and delta. Although in nature there might be more than one wolf in each category, it is considered that there is only one solution belong to each class in GWO for the sake of simplicity.

The concepts of alpha, beta, delta and omega are illustrated in Fig. 6. Note that the objective is to find the minimum in this search landscape. It may be seen in this figure that alpha is the closet solution to the minimum, followed by beta and delta. The rest of solutions are considered as omega wolves. There is just one omega wolf in Fig. 6, but there can be more.

In GWO, it is assumed that alpha, beta and delta are always the three best solutions obtained so far. The global optimum of optimization problems is unknown, so it has been assumed that alpha, beta and delta have good idea of its location, which is reasonable because they are the best solutions in the entire population. Therefore, other wolves should be obliged to update their positions as follows:

$$\mathbf{X}(t+1) = \frac{1}{3}\mathbf{X}_1 + \frac{1}{3}\mathbf{X}_2 + \frac{1}{3}\mathbf{X}_3 \quad (5)$$

where \mathbf{X}_1 and \mathbf{X}_2 and \mathbf{X}_3 are calculated with Eq. 6.

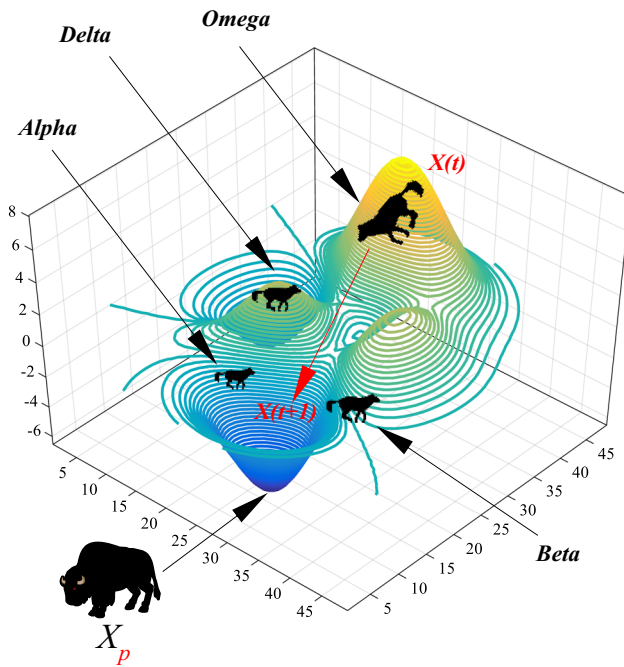


Fig. 6 How alpha, beta, delta and omega are defined in GWO

$$\begin{aligned} X_1 &= X_\alpha(t) - A_1 \cdot D_\alpha \\ X_2 &= X_\beta(t) - A_2 \cdot D_\beta \\ X_3 &= X_\delta(t) - A_3 \cdot D_\delta \end{aligned} \quad (6)$$

D_α , D_β and D_δ are calculated using Eq. 7.

$$\begin{aligned} D_\alpha &= |C_1 \cdot X_\alpha - X| \\ D_\beta &= |C_2 \cdot X_\beta - X| \\ D_\delta &= |C_3 \cdot X_\delta - X| \end{aligned} \quad (7)$$

2.3 GWO procedure

The GWO algorithm is a swarm intelligence algorithm, so it initiates the optimization process with a set of random solutions; there is a vector for every solution that maintains the values for the parameters of the problem. In each iteration, the first step is to calculate the objective value of each solution. Therefore, every solution is equipped with one variable to store its objective value.

In addition to the vectors and variables mentioned so far to save key data in memory when solving problems with GWO, there are three more vectors and three more variables. These vectors and variables store the locations and objective values value of alpha, beta and delta wolves in the memory. These variables should be updated prior to position updating process.

The GWO algorithm repeatedly updates the solutions using Eqs. 5–7. In order to calculate these equations, the

distance between the current solution and alpha, beta and delta should be calculated first using Eq. 7. The contributions of alpha, beta and delta in updating the position of the solution are then calculated using Eq. 6. Regardless of the objective value of solutions and their positions, the main controlling parameters of GWO (A , C and a) are updated prior to position updating.

2.4 Exploration and exploitation in GWO

Exploration and exploitation are two conflicting processes [8] that an algorithm might show when optimizing a given problem. In the exploration process, the algorithm tries to discover new parts of the problem search space by applying sudden changes in the solutions since the main objective is to discover the promising areas of the search landscape and prevent solutions from stagnating in a local optimum.

In exploitation, the main objective is to improve the estimated solutions achieved in the exploration process by discovering the neighbourhood of each solution. Therefore, gradual changes in the solutions should be made to converge towards the global optimum. The main challenge here is that exploration and exploitation are in conflict. Therefore, an algorithm should be able to address and balance these conflicting behaviour during optimization to find an accurate estimation of the global optimum for a given problem.

The main controlling parameter of GWO to promote exploration is the variable C . This parameter always returns a random value in the interval of $[0, 2]$. It changes the contribution of the prey in defining the next position. This contribution is strong when $C > 1$; the solution gravitates more towards the prey. Since this parameters provides random values regardless of the iteration number, exploration is emphasized during optimization in case of any local optima stagnation.

Another controlling parameter that causes exploration is A . The value of this parameter is defined based on a , which linearly decreases from 2 to 0. Due to the random components in this parameter, the range changes in the interval of $[-2, 2]$ for the parameter A . Exploration is promoted when $A > 1$ or $A < -1$, whereas there is emphasize on exploitation when $-1 < A < 1$.

As mentioned above, a good balance between exploration and exploitation is required to find an accurate approximation of the global optimum using stochastic algorithms. This balance is done in GWO with the decreasing behavior of the parameter a in the equation for the parameter A . This is illustrated in Fig. 7. This figure shows five line charts calculated for A . It may be seen that although this parameter is stochastic, it results in

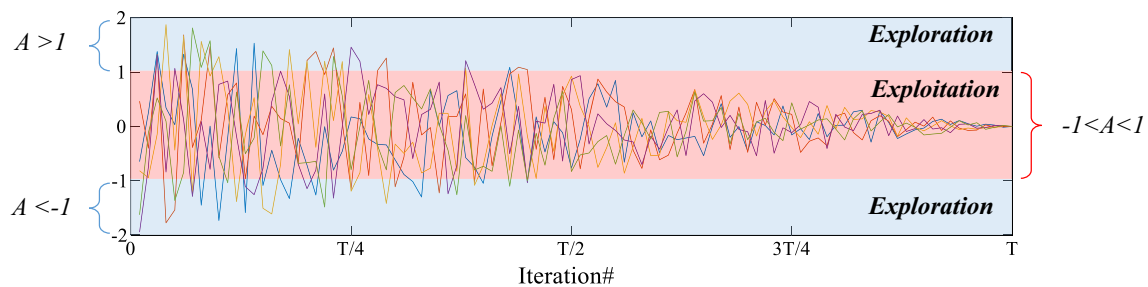


Fig. 7 Impact of A on exploration and exploitation. Note that the value of A when running GWO five times is given in this figure. It is evident that the parameter A fluctuates adaptively from the first to the last iterations, while the range is always in the interval of $[-2, 2]$. Considering Eq. 1, a wolf moves towards the prey when $-1 < A < 1$.

exploration in the first half of the iterations and then exploitation in the second half.

2.5 Illustrative example

What guarantees the convergence of the GWO algorithm is a proper balance between exploration and exploitation using the variable A . In addition, the three best solutions always guide other solutions towards the promising regions of the search space. This results in a high probability of improving the objective value of the population over the course of optimization. To show how GWO estimates the global optimum for optimization problems, Fig. 8 is given. This figure shows one unimodal and one multi-modal test functions solved by 20 solutions over 100 iterations.

This figure shows that the adaptive mechanism of the main controlling parameter (A) causes an interesting pattern in the trajectory of solutions. Inspecting the fourth column of Fig. 8, it can be seen that solutions face gradual changes proportional to the number of iterations. This

However, it moves away from (outwards) the prey when $A > 1$ or $A < -1$. In the former case, a grey wolf searchers (exploit) areas around the prey. In the latter case, a wolf explores other regions that are not in the vicinity of the prey

shows that GWO properly balances exploration and exploitations. The result of the proper exploration and exploitation can be seen in the search history (second column) of Fig. 8 as well. Note that the colour of each population is defined based on the iteration number, so the colour changes smoothly from a light grey to black. The global optimum estimated is also highlighted in red. It may be seen that the solutions discover the most promising regions of the search landscape and eventually converges towards the global optimum.

It was mentioned above that GWO is able to improve a random population. This can be observed in the fifth column of Fig. 8, in which the average fitness of solutions in the population tends to decrease and show an accelerated decline proportional to the number of iterations. Finally, the last column of this figure shows that the estimation of the global optimum becomes more accurate proportional to the number of iterations. The convergence curves are also accelerated in the last steps of optimization, which is due to

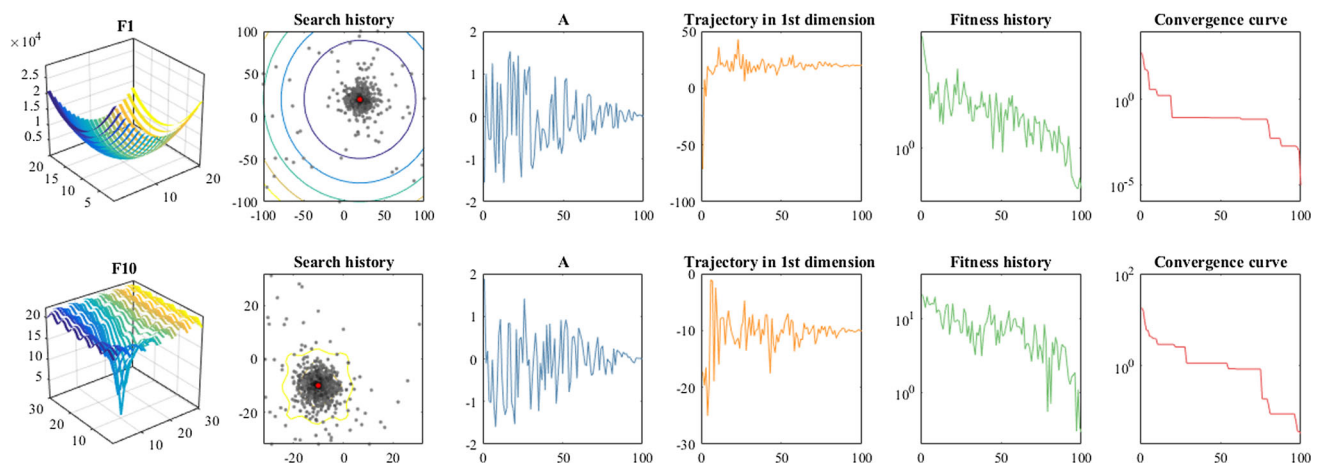


Fig. 8 Behaviour of GWO when solving a unimodal (F1) and a multi-modal (F10) test function. Search history of wolves in 100 iterations, the parameter A , trajectory of the first variable of the first wolf, average fitness history of all wolves, and the best wolf in each

iteration are visualized. The search pattern with good coverage can be seen in the search history. The impact of A on the trajectory is evident as well. Finally, the improvement in entire population and the best wolf can be observed in the last two columns

the low values of the parameter A after the first half of the optimization.

3 Recent GWO variants

Due to the complexity nature of real-world optimization problems, GWO have been modified to be inline with the search space of complex domains. Some modifications are done in the update mechanism since the GWO has some limitation in application to real-world problems. Other modifications are accomplished to improve the operations of GWO. Further improved versions apply the idea of hybridization to empower the exploration or exploitation sides of GWO. Other versions are formulated to handle the parallel computing platforms. In this section, the new versions of GWO are discussed in the following subsections.

3.1 Modified versions of GWO

The reviewed papers that aim at improving the performance of GWO are categorized into the following categories according to the type of modification that they proposed in the GWO:

- Updating mechanisms
- New operators
- Encoding scheme of the individuals.
- Population structure and hierarchy.

In the following, we highlight the main contributions that targeted each of these levels.

3.1.1 Updating mechanism

In this research direction, researchers targeted the improvement in the balance between exploration and exploitation processes which were described previously in Sect 2.4. To improve this balancing, two main sub-directions are identified: the first attempted to dynamically update the parameters of GWO, while the second proposed different strategies for updating the individuals. The main works of the category are discussed as follows.

In [85], the author studied the possibility of enhancing the exploration process in GWO by decreasing the value of a using an exponential decay function as given in Eq. 8, instead of changing it linearly. In Eq. 8, $iter$ denotes the current iteration, while $maxIter$ denotes the maximum number of iterations. They tested their approach using 27 benchmark functions and compared it with PSO, BAT algorithm (BA), CS and GWO. They claimed the modified version enjoys better exploration based on the results achieved.

$$a = 2 \left(1 - \frac{iter^2}{maxIter^2} \right) \quad (8)$$

Long et al. [59] also investigated adapting the parameter a nonlinearly using Eq. 9 where μ is nonlinear modulation index from the interval (0, 3). Their experimental results based on a number of constrained benchmark problems showed that better balancing between exploration and exploitation can be achieved using this nonlinear adaptation.

$$a = \left(1 - \frac{iter}{maxIter} \right) \cdot \left(1 - \mu \cdot \frac{iter}{maxIter} \right)^{-1} \quad (9)$$

Dynamic adaptation of a and C parameters was also proposed in [105]. However, the author used fuzzy logic to implement this dynamic adaptation.

Dudani and Chudasama [14] adopted a strategy for updating the position of the wolves based on incorporating a step size that is proportional to the fitness of the individual in the search space in the current generation as given in Eq. 10. The advantage of this strategy is that it has less parameters and it is not required to define the initial parameters. Based on a comparison of 21 benchmark functions, this version of GWO showed faster convergence and better results than the original GWO.

$$X_i^{t+1} = \left(\frac{1}{t} \right)^{\frac{|(bestf(t) - f_i(t)) / (bestf(t) - worstf(t))|}{1}} \quad (10)$$

where X_i^{t+1} is the step size of the i th dimension in t th iteration; $f(t)$ is the fitness value.

Malik et al. [73] adopted different approach for updating the positions of the individuals. Instead of using the simple average of the best individuals α , β and δ , they used a weighted average of these three. That is, each of the best individuals is assigned a weight calculated by multiplying its corresponding A and C . Their experiments showed superiority of the algorithm for multi-modal benchmark functions.

In [107], Rodríguez et al. proposed three different methods to update the positions of the omega wolves in the algorithm. The three methods are the weighted average, based on the fitness, and based on fuzzy logic. Although the performance with each method depends on the problem, they claimed a better performance for the fuzzy logic based method in majority of the tested benchmark functions.

3.1.2 New operators

This type of works focused on investigating the ability of improving the performance of GWO by integrating into it new operators like crossover or by utilizing a local search

algorithm. The main works in this direction of research are discussed as follows.

In an attempt to increase the diversity among the population in GWO, in [56], the authors proposed a modified version of the GWO that incorporates a simple crossover operator between two randomly different individuals. The aim of the crossover operator is to facilitate sharing information between pack mates. A comparison between GWO and their modified version based on six benchmark functions showed that the crossover operator enhanced the performance of GWO solution quality and convergence speed. This version of GWO was later applied by Chandr et al. [7] for selecting web services with optimal quality-of-service requirements.

Improving GWO with another type of operators called evolutionary population dynamics (EPD) was investigated by Saremi et al. [111]. EPD was applied in GWO to eliminate the worst individuals in the population and the reposition them around the leading wolves in the population, i.e. α , β and δ . The authors reported that the advantage of using this operator is that it improves the median of the population over the course of the iterations and it has positive influence of the exploration and exploitation processes. A comparison with GWO over 13 unimodal and multi-modal benchmark functions confirmed the advantage of such an operator.

Zhang and Zhou [139] integrated Powell optimization algorithm into GWO as a local search operator and called it PGWO. Powell's algorithm is a method for finding a minimum of a function which does not need to be differentiable, and no derivatives are taken. The algorithm performs successive bidirectional search along search vectors [98]. They tested PGWO using only seven unimodal and multi-modal benchmark functions. A comparison with GWO, GA, PSO, Adaptive Gbest-guided Gravitational Search Algorithm (GGSA), ABC and CS showed a an improvement in the performance over GWO and very competitive results compared to the other optimizers. It would be interesting to test the efficiency of PGWO on a larger number of benchmark functions including more sophisticated ones like rotated functions.

Mahdad and Srairi [72] combined GWO with pattern search algorithm as a local search for solving the security smart grid power system management at critical situations. Their results showed promising for this approach compared to many other optimization methods. However, it is still interesting to evaluate this GWO version on standard benchmark functions.

Zhou et al. [142] proposed combining GWO with chaotic local search tuning the parameters of the equivalent model of small hydro generator cluster. Their version of GWO showed noticeable improvement compared to PSO.

Other extensions of GWO are done in [107] which proposed a fuzzy hierarchical operator.

3.1.3 Different encoding schemes

In [70], a modification of GWO was proposed at the encoding scheme of the individuals. The authors used a complex-valued encoding method instead of the typical real-valued one. In this encoding, the genes in the individual have two main parts: an imaginary part and a real one. The authors argue that this presentation can expand the information capacity of the individual and increase the diversity of the population. In comparison with the classic GWO, ABC and GGSA, the complex-encoded GWO showed very competitive results based on 16 benchmark functions.

3.1.4 Modified population structure and hierarchy

GWO has a unique hierarchical population structure. As explained in Sect. 2, there are four different types of individuals; there types have only one individual and the rest is the fourth type. This distinguished structure has motivated some researchers to study the effect of proposing some modifications in the hierarchy. A notable work was carried out by Yang et al. [134], where a variant of GWO was proposed with different leadership hierarchy. The population in the variant is divided into two independent subpopulations: the first is called cooperative hunting group, while the second is called random scout group. The task of the scout grouped is to perform wide exploration process, while the task of the cooperative hunting group is to perform a deep exploitation. Unlike the classical GWO, δ wolves in the new leadership hierarchy hunter are divided into two types δ_1 that represent hunter wolves and δ_2 that represent scout wolves [134]. The authors of this variant applied it for tuning the parameters of interactive proportional-integral controllers of doubly fed induction generator-based wind turbine. They compared their results to those obtained by GA, PSO, GWO and MFO. Their results showed better fitness values and higher stability compared to the other optimizers.

3.2 Hybridized versions of GWO

Generally speaking, hybridization in the context of metaheuristics refers to combining between two or more algorithms in order to exploit the advantages and powerful features of each. In literature, GWO was hybridized with other metaheuristic algorithms as well. For example, Kamboj [48] proposed hybridizing GWO with PSO in a sequential fashion for optimizing a single-area unit commitment problem. The quality of the generated solutions

were competitive; however, this approach suffers slower convergence due to the long sequential execution time. Other works proposed hybridizing GWO with DE as in [47] and [143]. Both works showed very competitive results compared to PSO and DE.

3.3 Parallelism

Parallelism is an approach can be effectively applied in population-based metaheuristics where the population can be separated and divided into a number of subpopulations where each subpopulation can be evolved on a different machine processor. Parallelism can effectively reduce the execution time of the optimizer and enhance the quality of the solutions. A parallel version of GWO was implemented in [97]. In this implementation, the subgroups evolve independently and exchange best individuals every predefined number of iterations. A comparison with the original GWO shows that parallelism improved the performance significantly in terms of solution quality and running speed time over 4 different benchmark functions.

3.4 Multi-objective GWO

The GWO algorithm is able to solve problems with single objective. The multi-objective version of GWO (MOGWO) has been proposed in the literature to solve multi-objective problems. MOGWO utilizes the same mechanism to update the position of solutions. Due to the existence of multiple best solutions—so-called Pareto optimal solutions—in multi-objective problems, several modifications were made to GWO [84].

An archive was employed to maintain Pareto optimal solutions. The archive was not only a storage but also a selector of alpha, beta and delta. The archive controller selects a leader from the less populated regions of the Pareto optimal front estimated thus far and returns them as alpha, beta, or delta. This mechanism has been designed to improve the distribution (coverage) of Pareto optimal solutions stored in the archive across all objectives.

Another mechanism to improve the coverage of solutions was the archive clean-up process. When the archive becomes full, solutions should be removed to accommodate the new non-dominated solutions. MOGWO removes solutions in the most populated regions of the archive. So, the probability of improving the coverage of solutions in the archive increases in the next iterations. A conceptual example of selecting leaders from the least populated regions and removal of solutions from the most population regions (when the archive is full) are illustrated in Fig. 9.

The convergence behaviour of the MOGWO is similar to that of GWO due to the use of same position update mechanisms. Solutions face abrupt changes in the first half

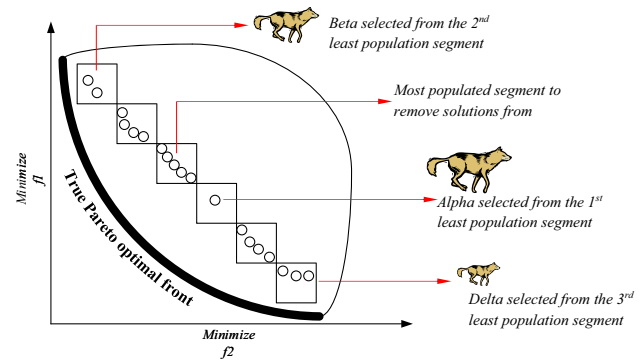


Fig. 9 Leader selection for the least populated regions and solution removal from the most population regions. In each iteration, MOGWO chooses leaders from the least populated region to drive other solutions towards it. This allows MOGWO to improve the distributions of Pareto optimal solutions estimated. By contrast, MOGWO removes solutions from most populated regions when the archive is full to accommodate adding new solutions in the least populated regions of the Pareto optima front obtained

of the iteration and gradual fluctuations in the rest. Due to the selection of alpha, beta and delta from the least populated regions, it is likely to have leaders from different regions for updating the position of a given solution. This degrades the convergence towards one best solution, yet it is required to maintain the distribution of solutions along the objectives.

4 Applications of GWO

Due to the impressive advantages of GWO, tremendous research applications from various crucial research domains have been tackled. These applications are categorized in this review paper into machine learning applications, engineering applications, wireless sensor network applications, environmental modelling applications, medical and bioinformatics application and image processing applications. In the following subsection, a comprehensive and exhaustive discussion is provided.

4.1 Machine learning applications

GWO has been applied in different machine learning applications. Most of these applications fall into main four categories: (1) feature selection, (2) training neural networks, (3) optimizing support vector machines and (4) clustering applications. These applications are in-depth discussed as follows.

4.1.1 Feature selection

Feature selection is one of the important processes in machine learning and data mining. The goal of feature selection is to reduce the number of features, select the most representative ones and to eliminate redundant, noisy and irrelevant features. The problem of searching for best set of features is considered as complex and difficult problem due to the extremely large search space when the number of features is large. In [21], Emary developed wrapper approach where a binary version of GWO combined with k -nearest neighbour (k -NN) as a fitness function to evaluate the candidate subsets of features. They compared their approach with GA and PSO over 8 benchmark datasets. Their GWO-based feature selection method showed faster convergence speed and higher classification accuracy. Later, in [22] Emary et al. extended their previous work and conducted a comprehensive study on utilizing two approaches of binary GWO in feature selection using different updating mechanisms. Following the same wrapper design with a k -NN fitness evaluator, experiments were conducted on 18 datasets. They compared the performance of GWO selection methods with GA and PSO. Their obtained results confirm the previous outcomes.

A different approach based on GWO for feature selection was proposed in [20]. This approach is consisted of two main consequent stages: the first is a filtered-base that utilizes mutual information equations as a fitness function, while the second is a wrapper-based that incorporates a classifier as an evaluator. Again, the k -NN algorithm was used as a classifier in the wrapper of the second stage. The experiments were conducted based on 8 datasets comparing GWO to PSO and GA. The developed approach showed promising results in terms of robustness and avoiding local minima.

It was noticed that in the previously mentioned three works, a very small population size was used ranging from 5 up to only 8.

Another binary GWO wrapper approach was applied in [129] for cancer classification on gene expression data. Different from the previous three works, in this one the author deployed a C4.5 decision tree algorithm as a fitness evaluator with k -fold cross-validation strategy and returning the accuracy rate as a fitness value. It is important to note here that their fitness depends only on the accuracy without incorporating number of features in it. This approach was tested based 10 micro-array cancer datasets and compared to different classifiers including support vector machine (SVM), self-organizing map (SOM), multilayer perceptron (MLP) networks and decision tree (C4.5). Their classification results were very competitive.

In [133], Yamany et al. proposed a feature reduction approach that used GWO to search the feature space for

subset of features that maximizes a rough set-based classification fitness function. Their fitness function considers the accuracy rate along with the number of selected features as in [22]. Yamany et al tested their approach based on 11 datasets and compared with conditional entropy-based attribute reduction (CEAR), discernibility matrix-based attribute reduction (DISMAR) and GA-based attribute reduction (GAAR). They used larger population size that the mentioned works which is 25 individuals. The proposed approach showed competitive results.

A simple modification of GWO to make it a binary version was also adopted by Medjahed et al. [75]. Their binary version was used for feature selection for hyperspectral band selection.

Another work in this direction was made by Li et al. [65] who proposed a binary version of GWO integrated as a component of a wrapper-based approach for feature selection. They used Kernel Extreme Learning Machine as a classifier for medical diagnosis problems

Medjahed et al. [75] converted the GWO to a binary form using a simple threshold. The authors applied their approach for hyperspectral band selection. Furthermore, [12] introduced GWO for feature selection in intrusion detection system.

4.1.2 Training neural networks

Artificial neural networks (ANNs) are information processing models inspired by the biological nervous systems. ANNs are widely applied in research and practice due to their high capability for capturing nonlinearity and dynamicity. However, the performance of ANNs is highly affected by their structure and connection weights. Traditionally, the efficiency of any new metaheuristic algorithm is investigated in optimizing the connection weights neural networks shortly after its release. GWO is no different. Mirjalili [77] applied GWO for training MLP, which is the most popular types of neural network. In his work, GWO was applied to optimize the weight and biases of a single hidden MLP network. The proposed training approach was compared with other well-known evolutionary trainers including: PSO, GA, ACO, ES and PBIL. The comparison results based on five classification and three function approximation datasets showed superiority of GWO in training MLP networks.

A similar approach to the previous one was conducted in [90]. The GWO-based training method was experimented based on three different datasets and compared to PSO, Gravitational Search Algorithm (GSA) and PSOGSA algorithms. The results showed very competitive performance for GWO and a trainer for MLP networks. An applied research of the same approach was conducted by Mohamed et al. [86], where a modified version of GWO

was used to train an MLP and applied to control firing angle of Static VAR compensator Controller (SVC) based on a high-dimensional input space. Their GWO-based MLP showed lower error values with faster convergence rates.

Radial basis function neural networks (RBFN) is another popular type of neural networks. A modified version of GWO was applied for a type of RBFNs called q -Gaussian radial basis functional-link nets (RBFLNs) in [92]. The authors compared the performance of GWO to PSO, teaching-learning-based optimization algorithm (TLBO) and GSA. The RBFLNs trained by GWO showed higher accuracy rates for different regression and classification problems.

4.1.3 Optimizing support vector machines (SVM)

SVM is considered as one of the powerful classifiers and regressors. SVM was founded by Vladimir Vapnik based on a strong mathematical foundation [126, 127]. To maximize the performance of SVM, two hyperparameters should be tuned; the error penalty parameter C and the kernel parameters. The problem is usually addressed by using a simple or exhaustive grid search. However, this method is not highly efficient due to the long running time needed for evaluating all possible combinations. Therefore, many researchers investigated optimizing these hyperparameters using metaheuristic algorithms. Recently, GWO was applied for tuning the hyperparameters of SVM in different publications. In [25], Eswaramoorthy et al. tuned gamma and sigma parameters in SVM for classifying intracranial electroencephalogram signals. The results showed higher accuracy rates compared to another classifier.

In [94], Mustaffa et al. applied GWO for tuning gamma and sigma hyperparameters of one of the variation of SVM called least squares SVM (LSSVM). In their work, GWO was compared to ABC and grid search algorithm with cross-validation. The GWO-LSSVM approach was experimented based on a time series data for real data of gold prices. The conducted experiments showed the results obtained by GWO-LSSVM are statistically significant compared to the other two approaches. Similarly, same authors applied this approach in [93] for forecasting time series natural gas prices. Promising results were obtained as well.

In [19], GWO was applied for tuning the penalty cost parameter and kernel parameters of SVM. The approach was tested for image classification. Different kernel functions were experimented in their study. However, there were no comparison with other approaches to evaluate the significance of the proposed approach. In [18], same

authors tuned the parameters of SVM using GWO for classifying Electromyography (EMG) signals.

4.1.4 Clustering applications

Clustering is a common machine learning and data mining task where the goal is to divide data instances into a number of groups that have similar characteristics in some sense [4]. Metaheuristic algorithms have been widely used and applied for clustering tasks. In the literature, most of the metaheuristic approaches for clustering are proposed as an alternative to the classical k -means algorithm which is one of the most famous clustering approaches. K -means algorithm highly depends on its initial selection of its centroids and it is highly probable that it will be trapped in a local minima. In this context, Kumar et al. [61] developed a clustering algorithm based on GWO in attempt to overcome the shortcomings of k -means algorithm. The basic idea is that each individual in GWO represents a sequence of a fixed number of centroids. For fitness evaluation, they used the sum of squared euclidean distance between each data point and the centroid of the cluster of the point. Their conducted experiments based on 8 datasets showed superior results of GWO compared to k -means algorithm and other metaheuristic algorithms.

Zhang and Zhou [139] proposed a GWO with Powell local optimization for clustering. They used same objective function and as the previously mentioned study. They compared their method with different metaheuristics including CS, ABC, PSO and GWO, and in addition to the classical k -means algorithm. The experiments based on 9 different datasets showed their modified GWO-based clustering approach outperformed the other algorithms in majority of the datasets. In a similar approach, Korayem et al. [58] also combined GWO for with k -means algorithms using same fitness function

Another approach was proposed by Yang and Lui [136] where GWO and K -means algorithms were hybridized in way that GWO's task was to select the initial centroids for the k -mean to overcome the dependency of the initial starting points.

4.2 Engineering applications

A very critical domain in the real-world optimization is engineering. It has a plethora of crucial applications that have directly influence the quality life of human beings. Interestingly, GWO has various adaptation to a wide variety of engineering applications. These applications are exhaustively and extensively discussed in this section which include design and tuning controllers, power dispatch problems, robotics and path planning, and many others as described below.

4.2.1 Design and tuning controllers

In control engineering, we have noticed an increased number of publications that investigate the application of GWO in tuning the parameters of controllers such as integral (I), proportional-integral (PI), proportional-integral-derivative (PID).

Li and Wang [66] used GWO for optimizing the parameters of a PI controller of a closed-loop condenser pressure control system. Their experimental results showed the effectiveness of GWO in comparison with other optimizers like GA and PSO.

Yada et al. [132] tuned the parameters of a conventional PID controller to levitate the metal ball of the magnetic levitation system. Compared to the classical Ziegler–Nichols (ZN) engineering tuning method, GWO showed better results.

Das et al. [10] experimented GWO for optimizing the parameters of a PID controller used for speed control of in a second order DC motor system. Better values of transient response specifications were obtained in comparison with PSO, ABC and ZN methods.

Tuning the parameters of PID controller parameters in DC motors was also investigated by Madadi and Motlagh [71]. In comparison with PSO, they showed that GWO can improve the dynamic performance of the system.

Sharma and Saikia [115] applied GWO for the optimizing secondary controller gains in automatic generation control. Their work showed better performance of GWO optimized PID controllers in terms of settling time, peak overshoot and magnitude of oscillations.

Gupta and Saxena [32] used GWO for tuning the parameters of automatic generation control for two area interconnected power system. For this problem, they compared the performance of GWO with GA, PSO, and GSA. Their experiments showed the superiority of GWO over the other optimizers with Integral Time Absolute Error as an objective function.

The parameters of automatic generation control (AGC) of interconnected three unequal area thermal system parameters were also tuned using GWO by Mallick et al. [74]. They compared the performance of GWO with bacteria foraging algorithm. GWO showed promising results.

In [101], Precup et al. tuned the parameters of Takagi–Sugeno proportional-integral-fuzzy controller for nonlinear servo systems. A comparison with PSO and GSA showed that no optimizer had clearly outperformed the other algorithms, but GWO has the advantage of simplicity and lower number of parameters to adjust. In same research are, similar approaches were followed also in [100, 101]. Noshadi et al. [96] also used GWO for finding the best design parameters of a PID fuzzy controller with some alternative proposed time-domain objective functions.

GWO showed very competitive results to PSO and better results than GA and imperialist competitive algorithm (ICA). Lal and Tripathy [62] utilized GWO for tuning a fuzzy PID controller for the automatic generation control study of two area nonlinear hydrothermal power system with a Thyristor controlled phase shifter. Their experiments showed that the developed system had higher effectiveness compared to DE, PSO and GWO PI-based controllers.

In [33], Gupta et al. conducted a comparison study on the performance of metaheuristics for tuning fuzzy PI controller for step set-point and trajectory tracking of reactor temperature in jacketed continuous stirred tank reactor. They compared the performance of GWO with backtracking search algorithm (BSA), DE and BAT algorithm (BA). GWO outperformed the other algorithms for both cases.

Razmjoooy et al. [104] used GWO to design an optimized linear quadratic regulator controller to control a single-link flexible manipulator. Compared to PSO, the authors showed in their experiments that GWO can improve the stability and performance of the manipulator.

For power point tracking, Yang et al. [134] proposed a new variant of GWO called group GWO for tuning the parameters of proportional-integral controllers of doubly fed induction generator-based wind turbine to achieve the highest power point tracking can be achieved along with improving fault-through capability. Mohanty et al. [87] described the design of a maximum power extraction algorithm using GWO for photovoltaic systems to work under partial shading conditions. Mohanty et al. another work in [88] proposed a hybridization between GWO and perturb and observe technique for extraction of maximum power from a photovoltaic with different possible patterns. Other design and tuning controllers applications are done in [128].

4.2.2 Power dispatch problems

The economic load dispatch (ELD) is a class of non-convex and highly nonlinear constrained optimization problem concern with finding an optimal load dispatch in order to operate and plan the current resources. This problem is a kind of optimization problem in which its complexity is increased based on the number of system units to be planned. It is concern with distributing the required electricity among the generating units in optimum way in order to minimize the fuel consumptions of each unit in accordance with a power balance equality constraints and power output inequality constraints. It is important to recall that a considerable amount of cost will be minimized if there is an optimum load dispatch is achieved. The application of ELD problem using grey wolf optimizer is initially proposed in [131]. Four classes of grey wolves such as alpha,

beta, delta and omega are used where the three main hunting operators (i.e. searching for prey, encircling prey and attacking prey) are adapted for such a problem. The GWO is evaluated using 20 generating units in economic dispatch, and the results show that the GWO is able to outperform those yielded by BBO optimizer, lambda iteration method (LI), Hopfield model-based approach (HM), CS, FA, ABC, neural networks training by ABC (ABCNN), quadratic programming (QP) and general algebraic modeling system (GAMS).

In the same year, another application for ELD combined with emission dispatch (EMD) using GWO is adapted in [118]. The main aim of this problem is to minimize the fuel cost and emission at the same time to determine the optimum power generation. Their system is tested using two systems containing 6 and 11 generating units of various constraints and load demand. Again, the comparative evaluation showed the superiority and the effectiveness of GWO. In [122], the reactive ELD problem is addressed by GWO where it used to determine the optimal combination of control variables such as voltages, tap changing transformers' ratios as well as the amount of reactive compensation devices. In order to evaluate their GWO, two power systems are used to show the convergence impacts: IEEE 30-bus system and IEEE118-bus system. Again, the results produced by GWO are recognized as a superior in comparison with those produced by comparative techniques. The ELD problem with ramp rate limits, valve point discontinuities and prohibited operating zones constraints was also proposed in [99]. Their system was implemented on four test systems having 10, 40, 80 and 140 units. Once again, the results of GWO were a very promising and proved that the GWO is a powerful technique for solving various ELD problems.

The non-convex and dynamic ELD problem version is tackled by GWO in [49]. Three kinds of power systems based on the system size are used for evaluation process. Their proposed system was compared with lambda iteration method, PSO, GA, BBO, DE, pattern search algorithm, hybrid of neural network (NN) and efficient particle swarm optimization (EPSO) method (NN-EPSO), fast evolutionary programming (FEP), classical EP (CEP), improved FEP (IFEP) and mutation FEP (MFEP). Comparative evaluation showed the superiority of GWO.

A combined heat and power dispatch (CHPD) problem was also tackled using GWO in [42, 43] and [44]. In addition to the normal constraints of ELD, the practical operational constraints like feasible operating regions of cogenerators, prohibited operating zones of thermal generators are also took in consideration. The main aim of GWO is to provide fuel cost savings and lesser pollutant emissions. Their proposed method was tested using 3 system types: 7-unit system, 24-unit system, and 11-unit

system. The simulation results showed that GWO performs better than the state-of-the-art methods in terms of solution quality.

In [41], the ELD problem is also addressed using a hybridized version of GWO. It hybridized with effective GA operators such as crossover and mutation for better performance. The proposed method was tested using four dispatch systems: 6, 15, 40 and 80 generators with valve point loading effect, prohibited operating zones and ramp rate limit constraints, with and without transmission losses. The comparative evaluation revealed that the hybrid GWO was either matches or outperforms the other comparative methods.

Another problem similar to ELD called pre-dispatch of thermal power generating units with cost, emission and reserve pondered was tackled by adapted GWO [103]. The main aim of this problem is to minimize the total operating cost, emission level and maximize reliability under various prevailing constraints. The performance of the adapted GWO was verified using standard 10, 20, 40, 60, 80 and 100 unit systems. The results showed that the proposed GWO is very promising in solving the targeted problem in comparison with other state-of-the-art methods. Other power engineering problems solved by GWO include [2, 27, 130, 135].

4.2.3 Robotics and path planning

In robotics technologies, a multi-objective GWO approach was proposed by Tsai et al. for optimizing robot path planning [125]. Two objectives were used as minimization which handle the distance and path smoothness. They performed a number of simulations in different static environments. Simulation results showed that the proposed approach succeeded in providing the robot with the optimal path to reach its target without colliding with obstacles.

An interested work was made by Zhang et al. [141] when they proposed GWO for solving the Unmanned Combat Aerial Vehicle (UCAV) path planning problem. They addressed three cases of different dimension path planning problems. The goal is to find a safe path while avoiding the threats areas and minimizing fuel cost. The experiments and results showed the efficiency of GWO compared to many other metaheuristic algorithms.

In another work conducted by Korayem et al. in the field of vehicle routing [58], they combined GWO with *k*-means algorithm for solving the capacitated vehicle routing (CVR) problem which is a type of the vehicle routing problems. The proposed hybrid approach is utilized in the clustering phase of the cluster-first route-second method for solving the targeted CVR problem. Their results showed higher efficiency of the hybrid approach compared to the *k*-means algorithm.

4.2.4 Scheduling

In welding production, Lu et al. [69] proposed a multi-objective discrete GWO for optimizing a real-world scheduling case from a welding process. They considered as objectives to minimize makespan and the total penalty of machine load. The experiment and the statistical tests showed the significance of the obtained results compared to other multi-objective evolutionary algorithms like non-dominated sorting-based genetic algorithm (NSGA-II) and improving the strength Pareto evolutionary algorithm (SPEA2).

Later on, same authors proposed a hybrid multi-objective approach based on GWO and GA for optimizing a multi-objective dynamic welding scheduling problem [68]. They considered three objectives which are minimizing the makespan, machine load and instability. Their approach was applied for a real welding scheduling problem from a company in China. A comparison with NSGA-II, SPEA2 and MOGWO showed better results in terms of the three objectives.

Another notable work in this type of applications was carried out by Komakia and Kayvanfar [57]. They proposed the application of GWO for scheduling a two-stage assembly flow shop problem. The goal was to find the optimal jobs sequence such that completion time of the last processed job is minimized. They compared GWO to PSO and cloud theory-based simulated annealing (CSA). GWO showed very promising results.

Rameshkumar et al. [102] proposed a real-coded GWO for optimizing short-term power system unit commitment schedule of a thermal power plant. In their work, the real-coded implementation of problem variables used to easily handle the operational constraints. Their approach showed a consistency for finding good unit commitment schedule within reasonable execution time. They conducted a comparison study and showed that the proposed GWO-based approach can outperform many existing methods. The author mentioned that their approach will be applied for long-term thermal power scheduling problems as well. In addition, optimal scheduling workflows in cloud computing environment using Pareto-based GWO have been investigated in [54].

4.2.5 Other engineering applications

Other engineering applications of GWO covers wide spectrum of different problems. More power and energy applications include the following main works: Hadidian-Moghaddam et al. [36] used GWO for optimizing the sizing of a hybrid photovoltaic/Wind system considering the reliability model. The objective function was set to minimize the system total annual cost. Sharma et al. [114]

proposed GWO to minimize the operation cost of micro-grid considering optimum size of battery energy storage. GWO showed high efficiency compared to other optimizers.

Sangwan et al. [109] used GWO for optimizing battery parameters for charging and discharging characteristics based on a model of single resistance-capacitance electrical circuit. In their work, GWO showed higher consistency and accurate estimation compared to GA and PSO. Mahdad and Srairi [72] proposed a combination of GWO with pattern search algorithm for solving the security smart grid power system management at critical situations.

Shakarami and Davoudkhani [113] optimized wide-area power system stabilizer parameters using GWO as a multi-objective problem with inequality constraints.

Zhou et al. [142] proposed combining GWO with chaotic local search tuning the parameters of the equivalent model of small hydro generator cluster. Their version of GWO showed noticeable improvement compared to PSO.

Sultana et al. [123] applied GWO for optimizing multiple distributed generation allocation in a 69-bus radial distribution system. The problem was tackled a multi-objective with a goal to minimize the reactive power losses and voltage deviation with a set of power system constraints. GWO was competitive compared to BA and GSA. However, they noted that GWO is slower than BA in terms of computational time.

For partial discharge modeling, Dudani and Chudasama [14] applied an adaptive version of GWO for localization of PD source using acoustic emission technique.

In power systems, El-Gaafary et al. [17] applied GWO for optimizing the allocation of STATCOM devices on a power system grid with a objective functions to minimize load buses voltage deviations and power system loss. Their experiments were conducted for IEEE 30-bus power systems.

In power flow problems, El-Fergany and Hasanien [16] applied GWO for solving the optimal power flow problem. They experimented four objective functions based on the standard IEEE 30-bus system. They reported competitive results for GWO compared to DE.

In photonics technologies, Chaman-Motlagh [6] proposed GWO for designing photonic crystal filters in attempt for finding high-performance designs. They handled the problem as an optimization problem of single-objective function.

Another work in the field, Karnavas et al. utilized GWO for optimizing the design of a radial flux surface permanent magnet synchronous motors. According to their conducted experiments and obtained results, GWO showed more efficient results and faster convergence speed.

In direct current (DC) motors, Karnavas and Chasiotis [51] optimized the parameters of an unknown permanent

magnet DC coreless micro-motor using GWO. They compared the performance of GWO for this problem with GA. They claimed that GWO could be faster than GA in estimating the parameters of the motor. The ensemble decision-based multi-focus image fusion using binary genetic GWO in camera sensor networks is also studied in [121].

In chemical engineering, Sodeifian et al. [117] used GWO for finding the optimal operating conditions as a part of a proposed methodology for the application of supercritical carbon dioxide to extract essential oil from *Cleome coluteoides* Boiss.

In quality and reliability engineering, Kumar and Pant [60] applied GWO for optimizing two complex reliability optimization problems: complex bridge and life-support system in a space capsule. They compared their results with other metaheuristics such as CS, PSO, ACO, and simulated annealing. GWO showed very competitive results.

In machined and manufacturing, Khalilpourazari et al. [55] applied GWO for tuning the parameters of a mathematical model of a multi-pass milling process which is a nonlinear multi-constrained problem. The GWO approach showed excellent results in terms of minimizing production time when compared to other approaches from literature and to other new metaheuristics.

In cellular networks, Ghazzai et al. [30] proposed approach based on GWO for solving the cell planning problem for the fourth-generation and long-term evolution (4G-LTE) cellular networks. The task of GWO is to find the optimal base station locations that satisfy problem constraints. However, their experiments showed that PSO outperformed GWO for this investigated problem.

In structural optimization, GWO was used by Bhensdadia and Tejani [5] for optimizing planer frame design. Their objective was to produce minimum weight planer frame while maintaining the material strength requirements. Their conducted experiments showed that GWO can find higher-quality designs than other metaheuristics.

In the field of antennas and electromagnetics, Saxena and Kothari [112] introduced different examples for the application of GWO for linear antenna array optimization for optimal pattern synthesis. The results showed that GWO was able to achieve good improvement compared to other metaheuristics.

In civil engineering, Gholizadeh [31] proposed a modified version of GWO named as Sequential GWO (SGWO) for optimizing the design of double layer grids considering nonlinear behaviour. Experiments based on two illustrative examples show the efficiency of the (SGWO) compared to GWO, Harmony Search (HS) and FA.

4.3 Wireless sensor network

Tackling the coverage problem in WSN, Shieh et al. [9] proposed a variation of GWO called Herds GWO (HGWO) for optimizing sensor coverage WSNs. The objective function of their approach considers coverage overlaps and holes of deploying WSN. Compared to GA and the classical GWO, the HGWO showed higher capability in finding quality solutions in terms of good coverage within reasonable computational time.

In WSN routing, Al-Aboody and Al-Raweshidy [1] proposed a three-level hybrid clustering routing protocol algorithm (MLHP) based on GWO for wireless sensor networks. The goal of the algorithm is to extend the network lifetime. The task of GWO in their implementation is the probabilistic selection of cluster heads in level two of the network. Their experiments showed that the throughput of the proposed approach is higher than for the other known algorithms.

In [28], Fouad et al. targeted the localization problem in WSNs. The authors proposed a sink node localization approach based on GWO. The objective function was set to find the nodes with high number of neighbours, and their total residual energies are high. Their GWO-based approach was evaluated under different scenarios of different networks' capacities. Their simulation results also confirmed the efficiency of GWO in terms of time complexity and energy cost.

Localization was also investigated by Nguyen et al. [39]. The authors proposed a multi-objective GWO for solving the node localization problem. The author pointed out that the multi-objective approach can be more efficient in solving the problem than the single-objective approach. They utilized two objective functions which are the distance of nodes and the geometric topology. They showed that the proposed approach can effectively reduce the average localization error.

4.4 Environmental modeling applications

In environmental modeling, several studies have deployed GWO in optimizing forecasting and quality models; Sweidan et al. [124] proposed a hybrid classification model based on case-based reasoning (CBR) and GWO for water pollution assessment based on fish gills microscopic images. The role of GWO was to select CBR suitable matching and similarity measures along with performing feature selection.

Malik et al. [89] proposed a constrained mathematical model and integrated a weighted distance-GWO for reducing the amount of pollutants generated by thermal power plants in Delhi, India. The proposed approach

showed high efficiency compared to the classical GWO and genetic programming.

Niu et al. [95] proposed a model for short-term daily PM_{2.5} concentration forecasting for Harbin and Chongqing cities in China. Their model was based on a decomposition-ensemble methodology. In this method, GWO was applied for optimizing the parameters of support vector regressor (SVR). The performance of the SVR optimizer by GWO had lower error rates for both cities compared to other optimizers including CS, DE, PSO and grid search.

A different application was conducted by Song et al. [119] who proposed a surface wave dispersion curve inversion scheme based on GWO. GWO showed very competitive results when compared to other optimizers like GA and PSO-GSA. The authors recommended the application of GWO for parameter estimation in surface waves.

4.5 Medical and bioinformatics application

GWO has been deployed in different approaches for various medical and bioinformatic applications. For example in [65], Li et al. used a binary GWO for feature selection with extreme learning machine (ELM) classifier for two medical diagnosis problems: Parkinson's disease diagnosis and breast cancer diagnosis. High-quality solutions were obtained.

In Bioinformatics, Jayapriya and Arock [45] utilized a parallel version of GWO for multiple sequences aligning problem. Their approach showed efficiency in terms of time complexity. In the same research area, the same authors proposed a variation of GWO for pairwise molecular sequence alignment. They deployed a new fitness function that returns maximum matched counts for new possible molecular sequences [46].

In medical diagnosis systems, Mostafa et al. [91] proposed a combination between GWO and statistical image of liver and simple region growing for liver segmentation in CT images. Their approach showed high accuracy rates. Other medical-based GWO problems are tackled in [108].

In [18], Elhariri combined GWO with SVM for EMG signal classification which has a wide range of clinical and biomedical applications. GWO was utilized to tune the hyperparameters of the SVM. The proposed approach showed promising results compared to other metaheuristics like GA, ACO, ES and PSO.

4.6 Image processing

In the field of image thresholding, Li et al. [64] addressed the problem of multi-level image thresholding problem. The authors proposed a modified discrete variation of GWO for optimizing fuzzy Kapur's entropy as an objective function to obtain a set of thresholds. Based on experiments

using a set of benchmark images, the proposed approach showed a capability of improving image segmentation quality.

For template matching problems, Zhang and Zhou [140] proposed a hybridization between GWO and lateral inhibition for solving complex template matching problems. In this context, the problems concern recognizing predefined template images in a source image. The comparative experiments showed that better solutions for template matching can be obtained compared to other recent metaheuristics.

For hyperspectral image classification, Medjahed et al. [75] proposed a GWO-based framework for band selection and to reduce the dimensionality of hyperspectral images with a goal not to reduce the classification accuracy of the image. The authors formulated the problem as a combinatorial optimization problem. Their objective function combined between classification accuracy and class separability measures. Evaluation results based on three hyperspectral images showed satisfactory results compared to other feature selection methods. Other GWO-based image processing problems were presented in [53, 63] where multi-level image thresholding was tackled and segmented.

5 Open source software of GWO

As mentioned in [133], the GWO algorithm is based on a very simple concept. GWO can be implemented in a few lines of code using simple mathematical operators which makes the algorithm computationally efficient. This helped in promoting the algorithm and made its popularity exceeds many other metaheuristics proposed in the same period.

Directly after the publishing the main paper of GWO [83], S. Mirjalili released a demo Matlab version of GWO.

Another implementation of GWO was presented by Gupta et al. [34]. The authors described their implementation of GWO as a toolkit in LabVIEW. LabVIEW is an integrated development environment for designing measurements, tests and control systems and applications. They compared the performance of their implementation with DE which is provided by the environment. Their comparison study was conducted based on 9 benchmark functions. The results showed faster convergence for GWO and better solutions compared to DE. Their results encourages engineers to apply GWO in different optimization problems in LabVIEW environment.

In an attempt to reach more researchers from the open source community, Faris et al. implemented GWO as part of EvoloPy [26]. EvoloPy is an open source optimization framework written in Python. The framework contained a number of recent metaheuristic algorithms. In their paper,

the authors showed that the running time of the implemented metaheuristics including GWO is faster than their implementation in Matlab for large-scale problems.

6 Assessment and evaluation of GWO

As discussed above, the GWO algorithm has been widely used to solve a variety of problems since proposal. The simple inspiration, little controlling parameter and adaptive exploratory behaviour are the main reasons of the success of this algorithm. Similarity to other stochastic optimization algorithms, however, it has a number of limitations and suffers from inevitable drawbacks.

The main limitation comes from the NFL theorem, which states that no optimization algorithm is able to solve all optimization problems. This means that GWO might require modification when solving some real-world problems. Another limitation is the single-objective nature of this algorithm that allows it to solve only single-objective problems. It should be equipped with special operators and mechanism to solve binary (combinatorial), continuous, dynamic, multi-objective and many-objective problems.

The main drawback of GWO is the low capability to handle the difficulties of a multi-modal search landscape, as it seems that all three alpha, beta and gamma wolves tend to converge to the same solution. Adding more random components to mutate the solutions during optimization will increase the chance of finding a global optimum when solving challenging multi-model problems.

The performance of the GWO algorithm degrades noticeably proportional to the number of variables. This is perhaps due to the entrapment of the initial population in a local solution when solving such problems. At the moment, there is not special operator to resolve such local optima stagnations in the literature.

The inventors of GWO conducted an extensive experiment and observed that considering four groups results in the best average performance on benchmark problems and a set of low-dimensional real-world case studies. However, considering more or less groups is required when solving medium- or large-scale challenging problems.

Last but not least, the fast convergence speed and accelerated exploitation leads to local solutions when solving problems with a large number of variables and local solutions. Mechanisms should be devised to decelerate the convergences and exploitations if the algorithm is trapped in local solutions. Adaptive mechanisms are good tools in this regard to tune the convergence speed proportional to the number of iterations of the quality of the best solution obtained so far.

7 Conclusions and possible research directions

This work provided the first literature review of the recently proposed GWO algorithm. The main inspiration, mathematical model, and analyses of this algorithm were discussed first. The performance of this algorithm was then investigated in terms of exploration and exploitation. The multi-objective version of this algorithm was presented briefly as well.

After the presentation and analysis, current works on parameter tuning, new operators, different encoding schemes and hybrids were reviewed and criticized. The applications of GWO in different fields were discussed as well: feature selection, training ANNs, classification, clustering, controller design, power dispatch problem, robotics and path planning, scheduling, wireless sensor network, environmental modeling, bioinformatics and image processing. This work also considered analysis of open source software for the GWO algorithm. Tables 1, 2, 3 and 4 show the review summaries of the related studies on GWO algorithm.

The search was done from five well-regarded scientific databases: Science Direct, SpringerLink, Scopus, IEEE and Google Scholar. The main keywords to find papers were “grey wolf optimizer”, “grey wolf optimiser”, “grey wolf optimization” and “GWO”. Figure 1 shows that the majority of publications on GWO were published by IEEE and Elsevier. It was also observed that more than 60% of GWO applications have been in the field of engineering.

Despite the popularity of GWO and recent advances, there are still several areas that need new or further works as follows:

- The GWO algorithm’s ability to handle a large number of variables and escaping local solutions when solving large-scale problems can be improved as the main drawbacks.
- A variety of memetic algorithms can be designed with hybridizing GWO and other current algorithms. The exploration and/or exploitation of GWO can be improved with employing operators from other algorithms.
- GWO divides the population into four groups, which has been proved to be an efficient mechanism to solve benchmark problems. Considering more or less groups with a diverse number of wolves in each can be considered as an interesting research area to improve the performance of GWO when solving challenging real-world problems.
- Dynamic optimization: there is currently no work in the literature to modify or use the GWO algorithm to solve dynamic problems. In a dynamic search space, the global optimum changes over time, so GWO should be

Table 1 Different modified versions of GWO algorithm

Modification level	Studies
Population structure and hierarchy	Mahdad et al. [72], Zhou et al. [142], Yang et al. [134]
Encoding scheme of the individuals	Luo et al. [70]
Updating mechanisms	Mittal et al. [85], Long et al. [67], Rodriguez et al. [105], Dudani et al. [14], Malik et al. [73], Rodríguez et al. [107]
New operators	Kishor et al. [56], Chandra et al. [7], Saremi et al. [111], Zhang et al. [139], Powell et al. [98]

Table 2 Hybrid, multi-objective and parallel versions of GWO algorithm

GWO Version	Studies
Hybrid GWO with PSO	Zhu et al. [143]
Hybrid GWO with DE	Jitkongchuen et al. [47], Kamboj et al. [48]
Parallel GWO	Pan et al. [97]
Multi-objective GWO	Mirjalili et al. [84], Emary et al. [20]

Table 3 Applications of GWO algorithm

Domain	Sub-domain	Studies
Machine learning	Feature selection	Emary et al. [20–22], Vosooghifard et al. [129], Yamany et al. [133], Medjahed et al. [75], Li et al. [65]
	Training neural networks	Mosavi et al. [90], Mohamed et al. [86], Muangkote et al. [92]
	Optimizing support vector machines	Eswaramoorthy et al. [25], Mustaffa et al. [93, 94], Elhariri et al. [19]
	Clustering	Kumar et al. [61], et al. Zhang [139], Yang et al. [136]
Engineering	Design and tuning controllers	Li et al. [66] Yadav et al. [132], Das et al. [10], Madadi et al. [71], Sharma et al. [115], Gupta et al. [32], Mallick et al. [74], Precup et al. [100, 101], Noshadi et al. [96], Lal et al. [62], Gupta et al. [33], Razmjoooy et al. [104], Yang et al. [134], Mohanty et al. [87, 88], Precup et al. [100]
	Power dispatch problems	Jayabarathi et al. [41], Pradhan et al. [99], Jayakumar et al. [42–44], Wong et al. [131], Rameshkumar et al. [103], Sulaiman et al. [122], Song et al. [118], Kamboj et al. [49]
	Robotics and path planning	Tsai et al. [125], Zhang et al. [141], Korayem et al. [58]
	Scheduling	Lu et al. [68, 69], Komaki et al. [57], Rameshkumar et al. [102], Lu et al. [68]
	Other engineering applications	Hadidian et al. [36], Sharma et al. [114], Sangwan et al. [109], Mahdad et al. [72], Shakarami et al. [113], Zhou et al. [142], Sultana et al. [123], El-Gaafary et al. [17], El-Fergany et al. [16], Chaman et al. [6], Karnavas and Chasiotis [51], Karnavas et al. [52], Sodeifian et al. [117], Kumar et al. [60], Khalilpourazari et al. [55], Ghazzai et al. [30], Bhensdadia et al. [5], Saxena et al. [112], Gholizadeh et al. [31] Sanjay et al. [110]
Networking	Wireless sensor network	Dao et al. [9], Al-Aboody et al. [1], Fouad et al. [28], Universty et al. [39]
Environmental applications	Modeling	Sweidan et al. [124], Mohideen et al. [89], Niu et al. [95], Song et al. [119]
Medical and bioinformatics		Lie et al. [65], Jayapriya et al. [45, 46], Mostafa et al. [91], Elhariri et al. [18]
Image processing		Li et al. [64], Zhang et al. [140], Medjahed et al. [75]

equipped with suitable operators (*e.g.* multi-swarm, repository, or performance measure) to solve such problems.

- Dynamic multi-objective optimization: there is no work in the literature to estimate a dynamically changing Pareto optimal front using the MOGWO algorithms.

Table 4 Open source software of GWO algorithm

Development framework	Studies
Matlab	Mirjalili et al. [83]
LabVIEW	Gupta et al. [34]
Python	Faris et al. [26]

Dynamic optimization in a multi-objective search space is very challenging and requires special consideration. High exploration of MOGWO makes it potentially able to discover different regions of a search space, but it needs modification to update the non-dominated solutions.

- Tuning the parameters of GWO has not been investigated well in the literature. Parameter tuning is important for all optimization algorithms when solving real-world problems. A valuable contribution would be to investigate alternative equations for the parameters a , A and C . Also, there is little work to define the weighting of α , β and δ in Eq. 5. These parameters define the contribution of each leader in the final solution and worth of investigation. Assigning priorities or weights to these solutions can be considered in this regard.
- Robust optimization: considering different uncertainties during optimization allows us to find reliable solutions. This is important when solving real problems with several uncertainties involved in inputs, outputs, objective functions and constraints. There is no work in the literature to investigate the performance of GWO in considering such perturbations and finding reliable solutions.
- Constrained optimization: despite a significant number of real-worlds problems that have been solved by GWO, there is no systematic work in the literature to investigate the best constraint handling methods for this algorithm. The majority of the current works use death penalty function, which is not effective when the search space has a large number of infeasible regions. Therefore, a good research direction is to equip GWO with different constraints handling techniques and investigate their performance.
- Binary optimization: there are several attempts in the literature to solve binary problems (mostly for feature selection). However, there is no systematic attempt to propose a general binary model for this algorithms. The investigating and integrating transfer functions or other operators to solve a wide range of binary problems (including combinatorial problems) are recommended.

The s-shaped and v-shaped transfer functions are worthy of investigation.

- Multi-objective optimization: although the MOGWO algorithm has been developed to solve multi-objective problems using the GWO algorithm, there are several gaps here that can be targeted. MOGWO uses an archive, but there are other mechanisms to solve multi-objective problems in the literature. Therefore, it is recommend to using different operators (*e.g.* non-dominated sorting, niching, archive, aggregation methods) to solve multi-objective optimization problems.
- Many-objective optimization: the current multi-objective variants of GWO use Pareto optimal dominance to compare solutions. As mentioned above, an archive stores non-dominated solutions. The archive is reasonably efficient for problems with up to four objectives. However, for many-objective problems, the archive becomes full quickly due to the large number of non-dominated solutions. Therefore, there should be spacial operators to solve such problems (*e.g.* hyper volume, modified Pareto optimal dominance and niching); a promising research avenue.
- Other means of opening venues for large-scale optimization problems [35] can be further investigated. The GWO also lacks of publications related to the theoretical analysis of its optimization framework. This should be addressed to provide more solid and in-depth understanding to the current GWO version and to answer the question of why GWO performs better than others. Therefore, a detailed and comprehensive theoretical analysis of the main properties of the GWO algorithm is highly recommended. Such analysis includes the study of the population structure, parameters, fitness landscapes. For example, why there are only three leaders in the algorithm? And how the performance will be affected if this number is increased or decreased.
- Other possible venue for interested audience is the parameter-free GWO. As aforementioned, GWO has two main parameters that control the search to achieve the correct trade-off between exploration and exploitation search aspects. These two parameters are set in advance and remain constant during the search. For naive users who are not familiars of using GWO in a proper way, the parameter setting is a very big dilemma. Therefore, parameter tuning of GWO can be urgently addressed in future as a very crucial need [106].

All the aforementioned reasons make GWO a possible and powerful candidate for utilization in different types of applications. In addition, this paper will draw a map for the researchers and practitioners who are currently working or

will work in this area by guiding them on how the GWO algorithm can be applied to different problems. Therefore, solving important complex problems through the utilization of GWO could be more possible.

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interest regarding the publication of this paper.

References

- Al-Aboody NA, Al-Raweshidy HS (2016) Grey wolf optimization-based energy-efficient routing protocol for heterogeneous wireless sensor networks. In: 2016 4th international symposium on computational and business intelligence (ISCBI). IEEE, pp 101–107
- Ali M, Elhameed MA, Farahat MA (2017) Effective parameters identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. *Renew Energy* 111:455–462
- Bäck T, Fogel DB, Michalewicz Z (1997) *Handbook of evolutionary computation*. Oxford University Press, New York
- Berkhin P (2006) A survey of clustering data mining techniques. Springer, Berlin, pp 25–71
- Bhensdadia V, Tejani G (2016) Grey wolf optimizer (GWO) algorithm for minimum weight planer frame design subjected to AISC-LRFD. In: *Proceedings of international conference on ICT for sustainable development*. Springer, pp 143–151
- Chaman-Motlagh A (2015) Superdefect photonic crystal filter optimization using grey wolf optimizer. *IEEE Photonics Technol Lett* 27(22):2355–2358
- Chandra M, Agrawal A, Kishor A, Niyogi R (2016) Web service selection with global constraints using modified gray wolf optimizer. In: 2016 international conference on advances in computing, communications and informatics (ICACCI). IEEE, pp 1989–1994
- Crepinsek M, Liu S-H, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. *ACM Comput Surv (CSUR)* 45(3):35
- Dao TK (2016) Enhanced diversity herds grey wolf optimizer for optimal area coverage in wireless sensor networks. In: *Genetic and evolutionary computing: proceedings of the tenth international conference on genetic and evolutionary computing*, November 7–9, 2016 Fuzhou City, Fujian Province, China, vol 536. Springer, p 174
- Das KR, Das D, Das J (2015) Optimal tuning of pid controller using gwo algorithm for speed control in dc motor. In: 2015 international conference on soft computing techniques and implementations (ICSCTI). IEEE, pp 108–112
- Davis L (ed) (1991) *Handbook of genetic algorithms*. Van Nostrand Reinhold, New York
- Devi EM, Suganthi RC (2017) Feature selection in intrusion detection grey wolf optimizer. *Asian J Res Soc Sci Hum* 7(3):671–682
- Dorigo M, Gambardella LM (1997) Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Trans Evol Comput* 1(1):53–66
- Dudani AR, Chudasama K (2016) Partial discharge detection in transformer using adaptive grey wolf optimizer based acoustic emission technique. *Cogent Eng* 3(1):1256083
- Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: *Proceedings of the sixth international symposium on micro machine and human science*, 1995. MHS'95. IEEE, pp 39–43
- El-Fergany AA, Hassanien HM (2015) Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. *Electric Power Compon Syst* 43(13):1548–1559
- El-Gaafary AAM, Mohamed YS, Hemeida AM, Mohamed AAA (2015) Grey wolf optimization for multi input multi output system. *Univ J Commun Netw* 3(1):1–6
- Elhariri E, El-Bendary N, Hassanien AE (2016) A hybrid classification model for EMG signals using grey wolf optimizer and SVMs. In: *The 1st international conference on advanced intelligent system and informatics (AIS2015)*, November 28–30, 2015, Beni Suef, Egypt. Springer, pp 297–307
- Elhariri E, El-Bendary N, Hassanien AE, Abraham A (2015) Grey wolf optimization for one-against-one multi-class support vector machines. In: 2015 7th international conference of soft computing and pattern recognition (SoCPar). IEEE, pp 7–12
- Emary E, Yamany W, Hassanien AE, Snasel V (2015) Multi-objective gray-wolf optimization for attribute reduction. *Procedia Comput Sci* 65:623–632
- Emary E, Zawbaa HM, Grosan C, Hassanien AE (2015) Feature subset selection approach by gray-wolf optimization. Springer, Cham, pp 1–13
- Emary E, Zawbaa HM, Hassanien AE (2016) Binary grey wolf optimization approaches for feature selection. *Neurocomputing* 172:371–381
- Engelbrecht AP (2006) *Fundamentals of computational swarm intelligence*. Wiley, Hoboken
- Engelbrecht AP (2007) *Computational intelligence: an introduction*. Wiley, Hoboken
- Eswaramoorthy S, Sivakumaran N, Sekaran S (2016) Grey wolf optimization based parameter selection for support vector machines. *COMPEL-The Int J Comput Math Electr Electron Eng* 35(5):1513–1523
- Faris H, Aljarah I, Mirjalili S, Castillo PA, Merelo JJ (2016) Evolopy: an open-source nature-inspired optimization framework in python. In: *IJCCI 2016—proceedings of the 8th International joint conference on computational intelligence*, vol 1, pp 171–177
- Fathy A, Abdelaziz AY (2017) Grey wolf optimizer for optimal sizing and siting of energy storage system in electric distribution network. *Electric Power Compon Syst* 45:1–14
- Fouad MM, Hafez AI, Hassanien AE, Snasel V (2015) Grey wolves optimizer-based localization approach in WSNs. In: 2015 11th international computer engineering conference (ICENCO). IEEE, pp 256–260
- Gandomi AH, Alavi AH (2012) Krill herd: a new bio-inspired optimization algorithm. *Commun Nonlinear Sci Numer Simul* 17(12):4831–4845
- Ghazzai H, Yaacoub E, Alouini MS (2014) Optimized lte cell planning for multiple user density subareas using meta-heuristic algorithms. In: 2014 IEEE 80th vehicular technology conference (VTC2014-Fall). IEEE, pp 1–6
- Gholizadeh S (2015) Optimal design of double layer grids considering nonlinear behaviour by sequential grey wolf algorithm. *J Optim Civ Eng* 5(4):511–523
- Gupta E, Saxena A (2016) Grey wolf optimizer based regulator design for automatic generation control of interconnected power system. *Cogent Eng* 3(1):1151612
- Gupta P, Kumar V, Rana KPS, Mishra P (2015) Comparative study of some optimization techniques applied to jacketed cstr control. In: 015 4th international conference on reliability, infocom technologies and optimization (ICRITO)(trends and future directions). IEEE, pp 1–6

34. Gupta P, Rana KPS, Kumar V, Mishra P, Kumar J, Nair SS (2015) Development of a grey wolf optimizer toolkit in labview. In: 2015 international conference on futuristic trends on computational analysis and knowledge management (ABLAZE). IEEE, pp 107–113
35. Gupta S, Deep K, Chamola BP, Kumari P (2017) Performance of grey wolf optimizer on large scale problems. In: AIP conference proceedings, vol 1802. AIP Publishing, p 020005
36. Hadidian-Moghaddam MJ, Arabi-Nowdeh S, Bigdeli M (2016) Optimal sizing of a stand-alone hybrid photovoltaic/wind system using new grey wolf optimizer considering reliability. *J Renew Sustain Energy* 8(3):035903
37. Hansen N, Kern S (2004) Evaluating the CMA evolution strategy on multimodal test functions. In: International conference on parallel problem solving from nature. Springer, pp 282–291
38. Höhfeld M, Rudolph G (1997) Towards a theory of population-based incremental learning. In: Proceedings of the 4th IEEE conference on evolutionary computation. Citeseer
39. Hai Phong Private University (2016) Estimation localization in wireless sensor network based on multi-objective grey wolf optimizer. In: Advances in information and communication technology: proceedings of the international conference, ICTA 2016, vol 538. Springer, p 228
40. Jain AK, Mao J, Mohiuddin KM (1996) artificial neural networks: a tutorial. *Computer* 29(3):31–44
41. Jayabarathi T, Raghunathan T, Adarsh BR, Suganthan PN (2016) Economic dispatch using hybrid grey wolf optimizer. *Energy* 111:630–641
42. Jayakumar N, Subramanian S, Elanchezhian EB, Ganesan S (2015) An application of grey wolf optimisation for combined heat and power dispatch. *Int J Energy Technol Policy* 11(2):183–206
43. Jayakumar N, Subramanian S, Ganesan S, Elanchezhian EB (2015) Combined heat and power dispatch by grey wolf optimization. *Int J Energy Sect Manag* 9(4):523–546
44. Jayakumar N, Subramanian S, Ganesan S, Elanchezhian EB (2016) Grey wolf optimization for combined heat and power dispatch with cogeneration systems. *Int J Electr Power Energy Syst* 74:252–264
45. Jayapriya J, Arock M (2015) A parallel gwo technique for aligning multiple molecular sequences. In: 2015 International conference on advances in computing, communications and informatics (ICACCI). IEEE, pp 210–215
46. Jayapriya J, Arock M (2016) Aligning two molecular sequences using genetic operators in grey wolf optimiser technique. *Int J Data Min Bioinf* 15(4):328–349
47. Jitkongchuen D (2015) A hybrid differential evolution with grey wolf optimizer for continuous global optimization. In: 2015 7th international conference on information technology and electrical engineering (ICITEE). IEEE, pp 51–54
48. Kamboj VK (2016) A novel hybrid pso-gwo approach for unit commitment problem. *Neural Comput Appl* 27(6):1643–1655
49. Kamboj VK, Bath SK, Dhillon JS (2015) Solution of non-convex economic load dispatch problem using grey wolf optimizer. *Neural Comput Appl* 27:1–16
50. Karnaboga D (2005) An idea based on honey bee swarm for numerical optimization. Technical report, technical report-tr06, Erciyes University, engineering faculty, computer engineering department,
51. Karnavas YL, Chasiotis ID (2016) PMDC coreless micro-motor parameters estimation through grey wolf optimizer. In: 2016 XXII international conference on electrical machines (ICEM). IEEE, pp 865–870
52. Karnavas YL, Chasiotis ID, Peponakis EL (2016) Permanent magnet synchronous motor design using grey wolf optimizer algorithm. *Int J Electr Comput Eng (IJECE)* 6(3):2016
53. Khairuzzaman AKM, Chaudhury S (2017) Multilevel thresholding using grey wolf optimizer for image segmentation. *Expert Syst Appl* 86:64–76
54. Khalili A, Babamir SM (2017) Optimal scheduling workflows in cloud computing environment using pareto-based grey wolf optimizer. *Pract Exp Concurr Comput*. <https://doi.org/10.1002/cpe.4044>
55. Khalilpourazari S, Khalilpourazary S (2016) Optimization of production time in the multi-pass milling process via a robust grey wolf optimizer. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-016-2644-6>
56. Kishor A, Singh, PK (2016) Empirical study of grey wolf optimizer. In: Proceedings of fifth international conference on soft computing for problem solving. Springer, pp 1037–1049
57. Komaki GM, Kayvanfar V (2015) Grey wolf optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time. *J Comput Sci* 8:109–120
58. Korayem L, Khorsid M, Kassem SS (2015) Using grey wolf algorithm to solve the capacitated vehicle routing problem. In: IOP conference series: materials science and engineering, vol 83. IOP Publishing, p 012014
59. Koza JR (1992) Genetic programming: on the programming of computers by means of natural selection. MIT Press, Cambridge
60. Kumar A, Pant S, Ram M (2016) System reliability optimization using gray wolf optimizer algorithm. *Qual Reliab Eng Int* 33:1327–1335
61. Kumar V, Chhabra JK, Kumar D (2017) Grey wolf algorithm-based clustering technique. *J Intell Syst* 26(1):153–168
62. Lal DK, Barisal AK, Tripathy M (2016) Grey wolf optimizer algorithm based fuzzy PID controller for AGC of multi-area power system with TCPs. *Procedia Comput Sci* 92:99–105
63. Li L, Sun L, Guo J, Qi J, Xu B, Li S (2017) Modified discrete grey wolf optimizer algorithm for multilevel image thresholding. *Comput Intell Neurosci* 2017:16
64. Li L, Sun L, Kang W, Guo J, Chong H, Li S (2016) Fuzzy multilevel image thresholding based on modified discrete grey wolf optimizer and local information aggregation. *IEEE Access* 4:6438–6450
65. Li Q, Chen H, Huang H, Zhao X, Cai ZN, Tong C, Liu W, Tian X (2017) An enhanced grey wolf optimization based feature selection wrapped kernel extreme learning machine for medical diagnosis. *Comput Math Methods Med* 2017:15. <https://doi.org/10.1155/2017/9512741>
66. Li SX, Wang JS (2015) Dynamic modeling of steam condenser and design of pi controller based on grey wolf optimizer. *Math Probl Eng* 2015:9
67. Long W, Liang X, Cai S, Jiao J, Zhang W (2016) A modified augmented Lagrangian with improved grey wolf optimization to constrained optimization problems. *Neural Comput Appl* 28(Suppl 1):S421–S438
68. Lu C, Gao L, Li X, Xiao S (2017) A hybrid multi-objective grey wolf optimizer for dynamic scheduling in a real-world welding industry. *Eng Appl Artif Intell* 57:61–79
69. Lu C, Xiao S, Li X, Gao L (2016) An effective multi-objective discrete grey wolf optimizer for a real-world scheduling problem in welding production. *Adv Eng Softw* 99:161–176
70. Luo Q, Zhang S, Li Z, Zhou Y (2015) A novel complex-valued encoding grey wolf optimization algorithm. *Algorithms* 9(1):4
71. Madadi A, Motlagh MM (2014) Optimal control of dc motor using grey wolf optimizer algorithm. *TJEAS Journal-2014-4-04/373-379* 4(4):373–79
72. Mahdad B, Srairi K (2015) Blackout risk prevention in a smart grid based flexible optimal strategy using grey wolf-pattern search algorithms. *Energy Convers Manag* 98:411–429
73. Malik MRS, Mohideen ER, Ali L (2015) Weighted distance grey wolf optimizer for global optimization problems. In: 2015

- IEEE international conference on computational intelligence and computing research (ICCIC). IEEE, pp 1–6
74. Mallick RK, Debnath MK, Haque F, Rout RR (2016) Application of grey wolves-based optimization technique in multi-area automatic generation control. In: International conference on electrical, electronics, and optimization techniques (ICEEOT)
75. Medjahed SA, Ait ST, Benyettou A, Ouali M (2016) Gray wolf optimizer for hyperspectral band selection. *Appl Soft Comput* 40:178–186
76. Mirjalili S (2015) The ant lion optimizer. *Adv Eng Softw* 83:80–98
77. Mirjalili S (2015) How effective is the grey wolf optimizer in training multi-layer perceptrons. *Appl Intell* 43(1):150–161
78. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. *Knowl-Based Syst* 89:228–249
79. Mirjalili S (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput Appl* 27(4):1053–1073
80. Mirjalili S (2016) Sca: a sine cosine algorithm for solving optimization problems. *Knowl-Based Syst* 96:120–133
81. Mirjalili S, Lewis A (2016) The whale optimization algorithm. *Adv Eng Softw* 95:51–67
82. Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Comput Appl* 27(2):495–513
83. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. *Adv Eng Softw* 69:46–61
84. Mirjalili S, Saremi S, Mirjalili SM, Coelho LS (2016) Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. *Expert Syst Appl* 47:106–119
85. Mittal N, Singh U, Sohi BS (2016) Modified grey wolf optimizer for global engineering optimization. *Appl Comput Intell Soft Comput* 2016:8
86. Mohamed AAA, El-Gaafary AAM, Mohamed YS, Hemeida AM (2015) Design static var compensator controller using artificial neural network optimized by modify grey wolf optimization. In: 2015 international joint conference on neural networks (IJCNN). IEEE, pp 1–7
87. Mohanty S, Subudhi B, Ray PK (2016) A new mppt design using grey wolf optimization technique for photovoltaic system under partial shading conditions. *IEEE Trans Sustain Energy* 7(1):181–188
88. Mohanty S, Subudhi B, Ray PK (2016) A grey wolf-assisted perturb observe MPPT algorithm for a PV system. *IEEE Trans Energy Convers* 32(1):340–347. <https://doi.org/10.1109/TEC.2016.2633722>
89. Malik MRS, Mohideen ER, Ali L, Raziuddin S (2016) Weighted distance grey wolf optimizer to control air pollution of delhi thermal power plant. *J Ind Pollut Control* 32(1). <http://www.icontrolpollution.com/articles/weighted-distance-grey-wolf-optimizer-to-control-air-pollution-of-delhi-thermal-power-plant-.php?aid=75885>
90. Mosavi MR, Khishe M, Ghamgosar A (2016) Classification of sonar data set using neural network trained by gray wolf optimization. *Neural Netw World* 26(4):393
91. Mostafa A, Fouad A, Houseni M, Allam N, Hassanien AE, Hefny H, Aslanishvili I (2016) A hybrid grey wolf based segmentation with statistical image for ct liver images. In: International conference on advanced intelligent systems and informatics. Springer, pp 846–855
92. Muangkote N, Sunat K, Chiewchanwattana S (2014) An improved grey wolf optimizer for training q-gaussian radial basis functional-link nets. In: 2014 international computer science and engineering conference (ICSEC), pp 209–214
93. Mustaffa Z, Sulaiman MH, Kahar MNM (2015) LS-SVM hyper-parameters optimization based on GWO algorithm for time series forecasting. In: 2015 4th international conference on software engineering and computer systems (ICSECS). IEEE, pp 183–188
94. Mustaffa Z, Sulaiman MH, Kahar MNM (2015) Training lssvm with gwo for price forecasting. In: 2015 international conference on informatics, electronics & vision (ICIEV). IEEE, pp 1–6
95. Niu M, Wang Y, Sun S, Li Y (2016) A novel hybrid decomposition-and-ensemble model based on ceemd and gwo for short-term pm 2.5 concentration forecasting. *Atmos Environ* 134:168–180
96. Noshadi A, Shi J, Lee WS, Shi P, Kalam A (2016) Optimal pid-type fuzzy logic controller for a multi-input multi-output active magnetic bearing system. *Neural Comput Appl* 27(7):2031–2046
97. Pan TS, Dao TK, Chu SC, et al (2015) A communication strategy for paralleling grey wolf optimizer. In: International conference on genetic and evolutionary computing. Springer, pp 253–262
98. Powell MJD (1977) Restart procedures for the conjugate gradient method. *Math Program* 12(1):241–254
99. Pradhan M, Roy PK, Pal T (2016) Grey wolf optimization applied to economic load dispatch problems. *Int J Electr Power Energy Syst* 83:325–334
100. Precup RE, David RC, Petriu EM (2017) Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity. *IEEE Trans Industr Electron* 64(1):527–534
101. Precup RE, David RC, Petriu EM, Szedlak-Stinean AI, Claudia-Adina B-D (2016) Grey wolf optimizer-based approach to the tuning of pi-fuzzy controllers with a reduced process parametric sensitivity. *IFAC-PapersOnLine* 49(5):55–60
102. Rameshkumar J, Ganesan S, Subramanian S, Abirami M (2015) Short-term unit commitment solution using real-coded grey wolf algorithm. *Aust J Electr Electron Eng* 13(1):54–66
103. Rameshkumar J, Ganesan S, Abirami M, Subramanian S (2016) Cost, emission and reserve pondered pre-dispatch of thermal power generating units coordinated with real coded grey wolf optimisation. *IET Gener Trans Distrib* 10(4):972–985
104. Razmjoo N, Ramezani M, Namadchian A (2016) A new lqr optimal control for a single-link flexible joint robot manipulator based on grey wolf optimizer. *Majlesi J Electr Eng* 10(3):53
105. Rodríguez L, Castillo O, Soria J (2016) Grey wolf optimizer with dynamic adaptation of parameters using fuzzy logic. In: 2016 IEEE congress on evolutionary computation (CEC). IEEE, pp 3116–3123
106. Rodríguez L, Castillo O, Soria J (2017) A study of parameters of the grey wolf optimizer algorithm for dynamic adaptation with fuzzy logic. In: Nature-inspired design of hybrid intelligent systems. Springer, pp 371–390
107. Rodríguez L, Castillo O, Soria J, Melin P, Valdez F, Gonzalez CI, Martinez GE, Soto J (2017) A fuzzy hierarchical operator in the grey wolf optimizer algorithm. *Appl Soft Comput* 57:315–328
108. Sahoo A, Chandra S (2017) Multi-objective grey wolf optimizer for improved cervix lesion classification. *Appl Soft Comput* 52:64–80
109. Sangwan V, Kumar R, Rathore AK (2016) Estimation of battery parameters of the equivalent circuit model using grey wolf optimization. In: 2016 IEEE 6th international conference on power systems (ICPS). IEEE, pp 1–6
110. Sanjay R, Jayabarathi T, Raghunathan T, Ramesh V, Mithulananthan N (2017) Optimal allocation of distributed generation using hybrid grey wolf optimizer. *IEEE Access* 5:14807–14818

111. Saremi S, Mirjalili SZ, Mirjalili SM (2015) Evolutionary population dynamics and grey wolf optimizer. *Neural Comput Appl* 26(5):1257–1263
112. Saxena P, Kothari A (2016) Optimal pattern synthesis of linear antenna array using grey wolf optimization algorithm. *Int J Antennas Propag* 2016:11
113. Shakarami MR, Davoudkhani IF (2016) Wide-area power system stabilizer design based on grey wolf optimization algorithm considering the time delay. *Electr Power Syst Res* 133:149–159
114. Sharma S, Bhattacharjee S, Bhattacharya A (2016) Grey wolf optimisation for optimal sizing of battery energy storage device to minimise operation cost of microgrid. *IET Gener Trans Distrib* 10(3):625–637
115. Sharma Y, Saikia LC (2015) Automatic generation control of a multi-area ST thermal power system using grey wolf optimizer algorithm based classical controllers. *Int J Electr Power Energy Syst* 73:853–862
116. Simon D (2008) Biogeography-based optimization. *IEEE Trans Evol Comput* 12(6):702–713
117. Sodeifian G, Ardestani NS, Sajadian SA, Ghorbandoost S (2016) Application of supercritical carbon dioxide to extract essential oil from cleome coluteoides boiss: Experimental, response surface and grey wolf optimization methodology. *J Supercrit Fluids* 114:55–63
118. Song HM, Sulaiman MH, Mohamed MR (2014) An application of grey wolf optimizer for solving combined economic emission dispatch problems. *Int Rev Model Simul (IREMOS)* 7(5):838–844
119. Song X, Tang L, Zhao S, Zhang X, Li L, Huang J, Cai W (2015) Grey wolf optimizer for parameter estimation in surface waves. *Soil Dyn Earthq Eng* 75:147–157
120. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Global Optim* 11(4):341–359
121. Sujatha K, Punithavathani DS (2017) Optimized ensemble decision-based multi-focus imagefusion using binary genetic grey-wolf optimizer in camera sensor networks. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-016-4312-3>
122. Sulaiman MH, Mustafa Z, Mohamed MR, Aliman O (2015) Using the gray wolf optimizer for solving optimal reactive power dispatch problem. *Appl Soft Comput* 32:286–292
123. Sultana U, Khairuddin Azhar B, Mokhtar AS, Zareen N, Beenish S (2016) Grey wolf optimizer based placement and sizing of multiple distributed generation in the distribution system. *Energy* 111:525–536
124. Sweidan AH, El-Bendary N, Hassanien AE, Hegazy OM, Mohamed A-K (2016) Grey wolf optimizer and case-based reasoning model for water quality assessment. In: *The 1st international conference on advanced intelligent system and informatics (AISi2015)*, November 28–30, 2015, Beni Suef, Egypt. Springer, pp. 229–239
125. Tsai PW, Dao TK, et al (2016) Robot path planning optimization based on multiobjective grey wolf optimizer. In: *International conference on genetic and evolutionary computing*. Springer, pp 166–173
126. Vapnik V (1995) *The nature of statistical learning theory*. Springer, New York
127. Vapnik V (1999) An overview of statistical learning theory. *IEEE Trans Neural Netw* 5:988–999
128. Verma SK, Yadav S, Nagar SK (2017) Optimization of fractional order pid controller using grey wolf optimizer. *J Control Autom Electr Syst* 28:1–9
129. Vosooghifard M, Ebrahimpour H (2015) Applying grey wolf optimizer-based decision tree classifier for cancer classification on gene expression data. In: *2015 5th international conference on computer and knowledge engineering (ICCKE)*. IEEE, pp 147–151
130. Wang L, Yue J, Su Y, Lu F, Sun Q (2017) A novel remaining useful life prediction approach for superbuck converter circuits based on modified grey wolf optimizer-support vector regression. *Energies* 10(4):459
131. Wong LI, Sulaiman MH, Mohamed MR, Hong MS (2014) Grey wolf optimizer for solving economic dispatch problems. In: *2014 IEEE international conference on power and energy (PECon)*. IEEE, pp 150–154
132. Yadav S, Verma SK, Nagar SK (2016) Optimized pid controller for magnetic levitation system. *IFAC-PapersOnLine* 49(1):778–782
133. Yamany W, Emary E, Hassanien AE (2016) New rough set attribute reduction algorithm based on grey wolf optimization. In: *The 1st international conference on advanced intelligent system and informatics (AISi2015)*, November 28–30, 2015, Beni Suef, Egypt. Springer, pp 241–251
134. Yang B, Zhang X, Yu T, Shu H, Fang Z (2016) Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine. *Energy Convers Manag* 133:427–443
135. Yang B, Zhang X, Yu T, Shu H, Zihao F (2017) Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine. *Energy Convers Manag* 133:427–443
136. Yang H, Liu J (2015) A hybrid clustering algorithm based on grey wolf optimizer and k-means algorithm. *J Jiangxi Univ Sci Technol* 5:015
137. Yang X-S (2009) Firefly algorithms for multimodal optimization. In: *International symposium on stochastic algorithms*. Springer, pp 169–178
138. Yang X-S, Deb S (2009) Cuckoo search via lévy flights. In: *World congress on nature & biologically inspired computing, 2009. NaBIC 2009*. IEEE, pp 210–214
139. Zhang S, Zhou Y (2015) Grey wolf optimizer based on Powell local optimization method for clustering analysis. *Discret Dyn Nat Soc* 2015:17
140. Zhang S, Zhou Y (2017) Template matching using grey wolf optimizer with lateral inhibition. *Optik* 130:1229–1243
141. Zhang S, Zhou Y, Li Z, Pan W (2016) Grey wolf optimizer for unmanned combat aerial vehicle path planning. *Adv Eng Softw* 99:121–136
142. Zhou J, Zhu W, Zheng Y, Li C (2016) Precise equivalent model of small hydro generator cluster and its parameter identification using improved grey wolf optimiser. *IET Gener Transm Distrib* 10(9):2108–2117
143. Zhu A, Xu C, Li Z, Wu J, Liu Z (2015) Hybridizing grey wolf optimization with differential evolution for global optimization and test scheduling for 3d stacked soc. *J Syst Eng Electron* 26(2):317–328