OPSEARCH
https://doi.org/10.1007/s12597-024-00785-x

THEORETICAL ARTICLE

A modified grey wolf optimization algorithm to solve global optimization problems

S. Gopi[1] · **Prabhujit Mohapatra**[1]

Accepted: 8 May 2024

Abstract

The Grey Wolf Optimizer (GWO) algorithm is a very famous algorithm in the field of swarm intelligence for solving global optimization problems and real-life engineering design problems. The GWO algorithm is unique among swarm-based algorithms in that it depends on leadership hierarchy. In this paper, a Modified Grey Wolf Optimization Algorithm (MGWO) is proposed by modifying the position update equation of the original GWO algorithm. The leadership hierarchy is simulated using four different types of grey wolves: lambda (λ), mu (μ), nu (ν), and xi (ξ). The effectiveness of the proposed MGWO is tested using CEC 2005 benchmark functions, with sensitivity analysis and convergence analysis, and the statistical results are compared with six other meta-heuristic algorithms. According to the results and discussion, MGWO is a competitive algorithm for solving global optimization problems. In addition, the MGWO algorithm is applied to three real-life optimization design problems, such as tension/compression design, gear train design, and three-bar truss design. The proposed MGWO algorithm performed well compared to other algorithms.

Keywords Meta-heuristic algorithms · Optimization problems · Statistical analysis · Benchmark functions · GWO · MGWO

List of symbols

$+/=/-$	The MGWO wins functions, ties functions, and loses functions
r_1 *and* r_2	Random vectors in [0,1]
λ	The best solution
b, b'	Linearly decreasing from 2 to 0

✉ Prabhujit Mohapatra
prabhujit.mohapatra@vit.ac.in; prabhujit.mohapatra@gmail.com

S. Gopi
gopi.s2020@vitstudent.ac.in; gopi3011998@gmail.com

[1] Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu 632 014, India

Published online: 01 June 2024

$D_\lambda, D_\mu, and\ D_\nu$	The distance between the current candidate wolves and the best three wolves
$L, b, L', and\ b'$	The coefficient vectors
Z, Z'	The grey wolf's position vector
$Z_\lambda, Z_\mu, and\ Z_\nu$	The position vectors of λ, μ, and ν wolves
Z_p, Z'_p	The position vectors of the prey
μ	The second best solutions
ν	The third best solutions
ξ	The other solutions
D	The dimension of the problems
M	The maximum number of iterations
n	Swarm size
t	The current iteration
ACO	Ant colony optimization
AIS	Artificial immune system
BBO	Biogeography-based optimization
CA	Cultural algorithm
CPA	Colony predation algorithm
DE	Differential evolution
ES	Evolutionary strategy
FHM	Fuzzy Hungarian method
FMODI	Fuzzy modified distribution method
GA	Genetic algorithm
GP	Genetic programming
GWO	Grey wolf optimizer
HHO	Harris hawks optimization
HOA	Horse herd optimization algorithm
IFHM	Intuitionistic fuzzy Hungarian method
IFMODI	Intuitionistic fuzzy modified distribution method
IFMZMCM	Intuitionistic fuzzy min-zero min-cost method
IFRM	Intuitionistic fuzzy reduction method
IGWO	Improved gray wolf optimization
MBO	Monarch butterfly optimization
MGWO	Modified grey wolf optimization algorithm
MGWOA	Mean grey wolf optimizer algorithm
MPA	Marine predators algorithm
NFL	No free lunch
PSO	Particle swarm optimization
SMA	Slime mould algorithm
SO	Snake optimization
SSA	Salp swarm algorithm
TSA	Tunicate search algorithm
WOA	Whale optimization algorithm
WSO	White shark optimization

1 Introduction

An optimization problem refers to a problem with multiple possible solutions, and the process of selecting the optimal solution among these options is known as optimization [1]. An optimization problem consists of decision variables, constraints, and an objective function [2, 3]. Advances in science and technology have led to more complicated and emerging optimization difficulties that require the use of relevant tools. There are two types of strategies for addressing optimization problems: deterministic and stochastic [4, 5]. Deterministic approaches, categorized as gradient-based and non-gradient-based, excel in solving linear, convex, and basic optimization problems. However, these approaches are ineffective for complicated, non-differentiable, nonlinear, non-convex, and NP-hard problems. These are the primary characteristics of optimization problems in real-life applications. Due to the limitations of deterministic methods, researchers have developed stochastic approaches like meta-heuristic algorithms [6]. Meta-heuristic algorithms are widely regarded as the most effective optimization algorithms due to their robustness, performance reliability, simplicity, and ease of implementation. Meta-heuristic algorithms are categorized into literary categories, like: (1) Evolutionary-based algorithms are based on evolutionary theory. (2) Swarm-based algorithms mimic the social behavior and decision-making of different groups. These algorithms rely on bio-community information and collaborative action to achieve certain goals. (3) Physics-based algorithms are influenced by natural physical principles. (4) Human behavior-based algorithms are inspired by human social behavior. (5) Hybrid and advanced algorithms use features from multiple optimization strategies to improve outcomes. Tables 1 and 2 show the classification of meta-heuristic algorithms in the literature.

The GWO algorithm is a meta-heuristic based on the hunting behavior and leadership hierarchy of grey wolves. It has been used to optimize key values in cryptography algorithms [70], time forecasting [71], feature subset selection [72], economic dispatch problems [73], optimal power flow problems [74], optimal design of double later grids [75], and flow shop scheduling problems [76]. Several algorithms have been developed to improve the convergence performance of the GWO algorithm, including binary GWO algorithm [77], parallelized GWO algorithm [78, 79], hybrid DE algorithm with GWO algorithm [80], hybrid GA algorithm with GWO algorithm [81], hybrid GWO algorithm using Elite Opposition Based Learning strategy and simplex method [82], Mean Grey Wolf Optimizer Algorithm (MGWOA) [83], integration of DE algorithm with GWO algorithm [84], and hybrid PSO algorithm with GWO algorithm [85]. The optimization problems have been simplified and solved using linear or integer programming techniques. Existing fuzzy and intuitionistic fuzzy optimization methods, such as FMODI, IFMODI, IFMZMCM, FHM, IFHM, and IFRM, can be complicated due to their numerous steps. Usually, fuzzy and intuitionistic fuzzy optimization problems are 1st translated to equivalent crisp optimization problems. Then second, to solve TORA software, crisp optimization

Table 1 Classification of meta-heuristic algorithms

Category	Algorithm name	References
Evolutionary-based algorithms	Differential evolution (DE)	[7]
	Genetic algorithm (GA)	[8]
	Genetic programming (GP)	[9]
	Evolutionary strategy (ES)	[10]
	Cultural algorithm (CA)	[11]
	Artificial immune system (AIS)	[12, 13]
	Biogeography-based optimization (BBO)	[14]
Swarm-based algorithms	Particle swarm optimization (PSO)	[15]
	Grey wolf optimization (GWO)	[16]
	Ant colony optimization (ACO)	[17]
	Marine predators algorithm (MPA)	[18]
	Whale optimization algorithm (WOA)	[19]
	Tunicate search algorithm (TSA)	[20]
	White shark optimization (WSO)	[21]
	Horse herd optimization algorithm (HOA)	[22]
	Snake optimization (SO)	[23]
	Slime mould algorithm (SMA)	[24]
	Monarch butterfly optimization (MBO)	[25]
	Colony predation algorithm (CPA)	[26]
	Harris hawks optimization (HHO)	[27]
	Moth search algorithm (MSA)	[28]
	RUNge Kutta optimization (RUN)	[29]
	Emperor penguin optimization (EPO)	[30]
	Orca predation algorithm (OPA)	[31]
	Artificial hummingbird algorithm (AHA)	[32]
	Chameleon swarm algorithm (CSA)	[33]
	Reptile search algorithm (RSA)	[34]
Physics-based algorithms	Simulated annealing (SA)	[35]
	Spring search algorithm (SSA)	[36]
	Gravitational search algorithm (GSA)	[37]
	Momentum search algorithm (MSA)	[38]
	Multi-verse optimizer (MVO)	[39]
	Equilibrium optimizer (EO)	[40]
	Water cycle algorithm (WCA)	[41]
	Archimedes optimization algorithm (AOA)	[42]
	Henry gas solubility optimization (HGSO)	[43]
	weIghted meaN oF vectOrs (INFO)	[44]
	Thermal exchange optimization (TEO)	[45]

Table 2 Classification of meta-heuristic algorithms

Category	Algorithm name	References
Human behavior-based algorithms	Teaching-learning based optimization (TLBO)	[46]
	Following optimization algorithm (FOA)	[47]
	Inperialist competitive algorithm (ICA)	[48]
	Poor and rich optimization (PRO)	[49]
	Group teaching optimization (GTO)	[50]
	League champion algorithm (LCA)	[51]
	Hunger games search (HGS)	[52]
	Political optimizer (PO)	[53]
	Brain storm optimization algorithm (BSOA)	[54]
	Dual-population social group optimization (DPSGO)	[55]
	Human eye vision algorithm (HEVA)	[56]
	Human mental search (HMS)	[57]
Hybrid and advanced algorithms	Modified whale optimisation algorithm	[58]
	Sine-cosine Harris hawks optimization	[59]
	Opposition-based learning cooking algorithm	[60]
	Hybrid binary ant lion optimizer	[61]
	Evolved opposition-based mountain gazelle optimizer	[62]
	Hybrid firefly algorithm with grouping attraction	[63]
	Improvised grey wolf optimization	[64]
	Modified hybrid GWO-SCA algorithm	[65]
	Improved moth-flame optimization	[66]
	Self-adaptive differential evolution	[67]
	Enhanced opposition-based grey wolf optimizer	[68]
	FROBL Aquila optimization algorithm	[69]

problems are transformed into equivalent linear programming problems. It uses branch and bound methods to solve problems. The use of fuzzy and intuitionistic fuzzy sets to solve optimization problems has been in the literature [86–91]. The methodology of estimating theory, statistical learning, and data mining is known as multivariate adaptive regression splines (MARS) [92]. Nowadays it is used in many different domains including science, technology, management, and economics [93–118]. Non-parametric regression analysis such as MARS does not require certain preconditions on the functional relationships between the explanatory and involved response variables. Since it automatically models interactions and non-linearities, it can be considered an extension of linear models [119]. MARS is unable to completely handle variable uncertainty despite all of its accomplishments.

Some of the issues/challenges in the existing literature. A fundamental challenge is that achieving global optimum value requires sluggish convergence and considerable computing overhead. A lot of algorithms lack a proper balance between exploration and exploitation abilities. A few algorithms converge prematurely to the local optimum, making them unsuitable for real-life engineering problems. Another drawback is that the algorithm has a large number of algorithm-specific parameters, and picking optimum values requires a high computing load.

The main research question in the study of meta-heuristic algorithms is whether there is still a need to propose new approaches despite the abundance of optimization algorithms created. In answer to this difficulty, the No Free Lunch (NFL) theorem [120] says that an algorithm's strong performance in dealing with a set of optimization problems does not ensure the same performance in other optimization problems. Therefore, claiming that an algorithm is optimal for all optimization applications is inaccurate. The NFL theorem encourages authors to propose innovative algorithms to solve optimization problems.

In this paper, a Modified Grey Wolf Optimization Algorithm (MGWO) is proposed by modifying the position update equation of the original GWO algorithm. The leadership hierarchy is simulated using four different types of grey wolves: lambda (λ), mu (μ), nu (ν), and xi (ξ). The MGWO algorithm addresses the shortcomings of leading wolves in GWO by enhancing their performance. The performance of the proposed algorithm has been evaluated on 23 benchmark functions, and the results were compared to popular meta-heuristic algorithms. MGWO is applied to three real-life engineering problems, and the results were compared to popular meta-heuristic algorithms.

The paper is organized as follows: Sect. 2 provides a brief introduction to the GWO algorithm. In Sect. 3 modified version of GWO named MGWO has been proposed. Results and discussion on CEC 2005 benchmark functions have been presented in Sect. 4. In Sect. 5 MGWO for solving real-life engineering problems are presented. Finally, in Sect. 6 we conclude the paper and suggest future works.

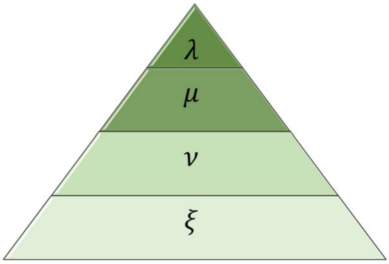
2 Grey wolf optimization algorithm (GWO)

The GWO algorithm is inspired by the hierarchy and hunting behavior of grey wolf groups. The method optimizes grey wolf populations by mathematically replicating the tracking, surrounding, hunting, and attacking processes. The grey wolf hunting procedure consists of three steps: social hierarchy stratification, encircling the prey, and attacking the prey.

2.1 Social hierarchy

Grey wolves are gregarious canids that live at the top of the food chain and have a tight social dominance structure. The best solution is indicated as the lambda (λ); the second-best solutions are marked as the mu (μ); the third-best solutions are marked as the nu (ν); and the other solutions are marked as xi (ξ). Figure 1 illustrates its dominating social hierarchy.

Fig. 1 Hierarchy of wolves



2.2 Encircling the prey of GWO

The wolves encircling approach around the prey is mathematically represented by the following equations:

$$Z(t+1) = Z_p(t) - L \times D, \tag{1}$$

$$D = |N \times Z_p(t) - Z(t)|, \tag{2}$$

where Z is the grey wolf's position vector, Z_p represents the position vectors of the prey, t represents the current iteration, and L and N are coefficient vectors.

The vectors L and N are calculated as follows:

$$L = 2 * r_1 \times b - b, \tag{3}$$

$$N = 2 \times r_2, \tag{4}$$

where b components are linearly decreasing from 2 to 0 over iterations and r_1, r_2 are random vectors in [0, 1].

2.3 Attacking the prey of GWO

Grey wolves can recognize possible prey locations, and the search is primarily carried out with the assistance of λ, μ, and ν wolves. The best three wolves (λ, μ, and ν) in the current population are preserved in each iteration, while the positions of other search agents are updated based on their position information. The following formulas are provided in this regard:

$$Z_1 = Z_\lambda - L_1 \times D_\lambda,\ Z_2 = Z_\mu - L_2 \times D_\mu,\ Z_3 = Z_\nu - L_3 \times D_\nu, \tag{5}$$

$$D_\lambda = |N_1 \times Z_\lambda - Z|,\ D_\mu = |N_2 \times Z_\mu - Z|,\ D_\nu = |N_3 \times Z_\nu - Z|, \tag{6}$$

$$Z(t+1) = \frac{Z_1 + Z_2 + Z_3}{3}, \tag{7}$$

In the above equation, Z_λ, Z_μ, and Z_ν are the position vectors of λ, μ, and ν wolves, respectively; the calculations of L_1, L_2, and L_3 are similar to L, while the calculations of N_1, N_2, and N_3 are similar to N. The distance between the current candidate wolves and the best three wolves is represented by D_λ, D_μ, and D_ν.

Figure 2 shows that the candidate solution eventually falls within the random circle formed by λ, μ, and ν. The other contenders then update their locations near the prey at random, guided by the current best three wolves. They begin searching for prey position information in a disorganized way before focusing on assaulting the prey.

3 A modified grey wolf optimization algorithm (MGWO)

A Modified Grey Wolf Optimization Algorithm (MGWO) was inspired by the GWO algorithm, which is already discussed in Sect. 2. Then the mathematical form of the MGWO algorithm is provided as follows:

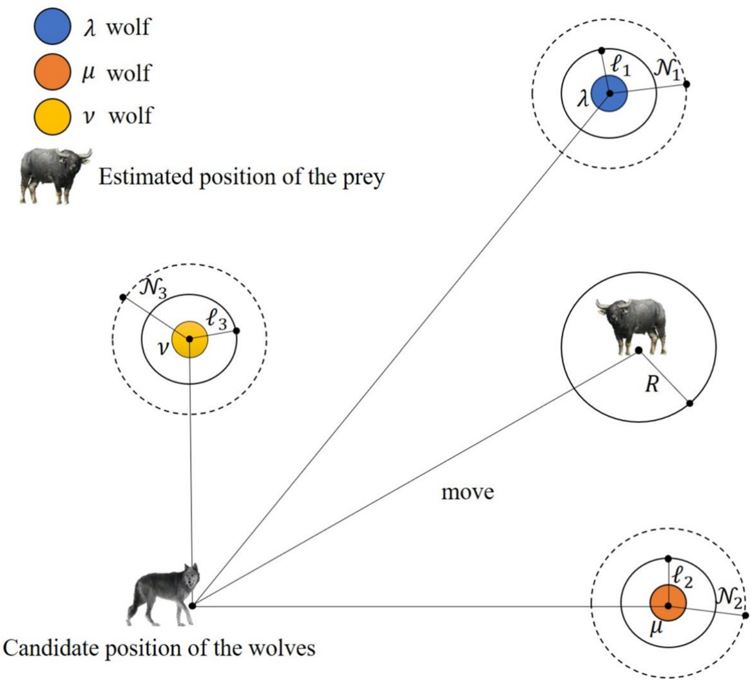


Fig. 2 Position updating in the gray wolf optimization (GWO)

3.1 Encircling the prey of MGWO

The wolves' encircling approach around the prey is mathematically described by providing the following equations as

$$Z'(t+1) = Z'_p(t) - L' \times D', \tag{8}$$

$$L' = 2 \times r_1 \times b' - b', \tag{9}$$

where Z' is the grey wolf's position vector, Z'_p represents the position vectors of the prey, t represents the current iteration, L' is coefficient vector, b' components are linearly decreasing from 2 to 0 over iterations and r_1 is random vectors in [0, 1].

$$D' = |N' \times Z'_p(t) - Z'(t)|, \tag{10}$$

$$N' = 2 \times r_2, \tag{11}$$

where N' is coefficient vector, b' components are linearly decreasing from 2 to 0 over iterations and r_2 is random vectors in [0, 1].

3.2 Attacking the prey of MGWO

Grey wolf attacking technique can be mathematically described by approximating the prey position using λ, μ, and ν solutions (wolves). As a result, by using this estimate, each wolf can update their positions by

$$Z'(t+1) = \frac{2}{3}Z'_1 + \frac{1}{4}Z'_2 + \frac{1}{12}Z'_3, \tag{12}$$

where Z'_1, Z'_2, and Z'_3 are calculated by using Eq. 13.

$$Z'_1 = Z'_\lambda - L'_1 \times D'_\lambda,\ Z'_2 = Z'_\mu - L'_2 \times D'_\mu,\ Z'_3 = Z'_\nu - L'_3 \times D'_\nu, \tag{13}$$

where L'_1, L'_2, L'_3, D'_λ, D'_μ, and D'_ν are calculated by using Eq. 14 and 15.

$$L'_1 = 2 \times r'_1 \times b' - b',\ L'_2 = 2 \times r'_2 \times b' - b',\ L'_3 = 2 \times r'_3 \times b' - b', \tag{14}$$

$$D'_\lambda = |N'_1 \times Z'_\lambda - Z'|,\ D'_\mu = |N'_2 \times Z'_\mu - Z'|,\ D'_\nu = |N'_3 \times Z'_\nu - Z'|, \tag{15}$$

where N'_1, N'_2, and N'_3 are calculated by using Eq. 16.

$$N'_1 = 2 \times r''_1,\ N'_2 = 2 \times r''_2,\ N'_3 = 2 \times r''_3, \tag{16}$$

The candidate solution eventually falls within the random circle formed by λ, μ, and ν. The other contenders then update their locations near the prey at random, guided by the current best three wolves. They begin searching for prey position information

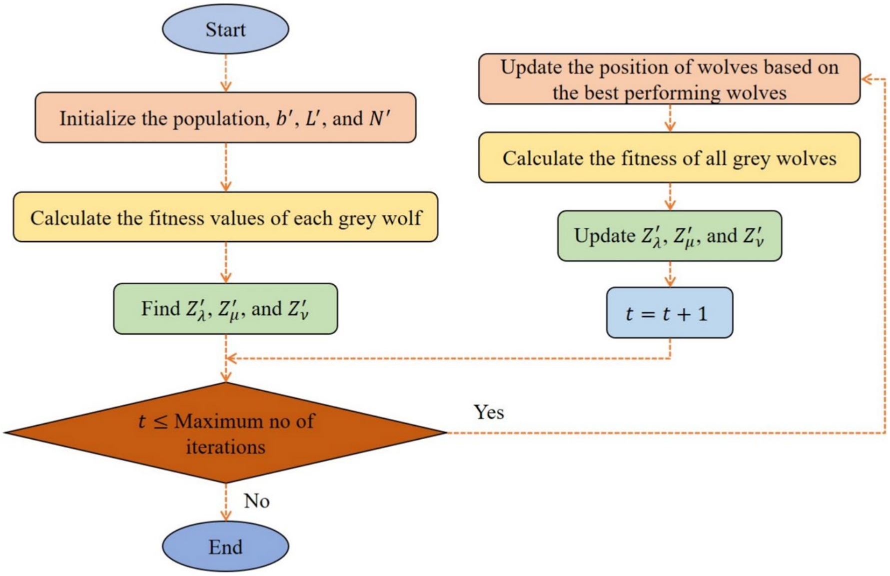


Fig. 3 Flowchart of MGWO

in a disorganized way before focusing on assaulting the prey. The pseudo code of the MGWO algorithm is presented in Algorithm 1. The flowchart of the MGWO is given in Fig. 3.

Algorithm 1 Pseudo-code of the MGWO algorithm.

```
Initialize the grey wolf population Z'_i (i = 1, 2, ··· , n).
Initialize the parameters b', L', and N'.
Calculate the fitness of each grey wolves.
Z'_λ= the first finest grey wolves.
Z'_μ= the second finest grey wolves.
Z'_ν= the third finest grey wolves.
while (t ≤ Maximum no of iterations) do
    for each grey wolves do
        Update the position of the current grey wolves by Equation 12.
    end for
    Update b', L', and N'.
    Calculate the fitness of each grey wolves.
    Update Z'_λ, Z'_μ, and Z'_ν.
    t = t + 1.
end while
Return Z'_λ.
```

3.3 Computational complexity

The computational complexity of the proposed MGWO algorithm depends on three main processes: initialization, evaluation of the fitness function, and updating each particle. The computational complexity of the basic process with a n particle is $O(n)$, and updating the MGWO mechanism is equal to $O(M \times n) + O(M \times n \times D)$, where M signifies the maximum number of iterations and D signifies the dimension of the problems. Therefore, the total computational complexity of the proposed MGWO is equals $O(n \times (M + M \times D + 1))$.

4 Results and discussion

In this section, to analyze the performance of the MGWO algorithm, seven uni-modal test functions, six multi-modal optimization functions, and 10 fixed-dimensional multi-modal optimization functions are selected. Table 4 lists these functions' precise expressions, dimensions, search space, and optimal values. The uni-modal test functions are represented by $F1$–$F7$, the multi-modal test functions by $F8$–$F13$, and the fixed dimensional multi-modal test functions by $F14$–$F23$. The uni-modal test function is primarily used to calculate the MGWO's convergence speed and accuracy of the solution. The primary purpose of the multi-modal test function is to gauge the MGWO's global surveying capability. To increase the experiment's accuracy, the six chosen algorithms use identical experimental parameters: swarm size (n = 30), dimension (D = 30), maximum number of iterations (M = 1000), each algorithm is run 30 times independently and the results are recorded. The values set for the control parameters of the competitor algorithms are given in Table 3. The

Table 3 The values set for the control parameters of the competitor algorithms

Algorithm	Parameter	Value
MGWO	l	[− 1, 1]
	r	[0, 1]
PSO	$C1$	1.5
	$C2$	2
	W	0.3
TSA	P_{min}	1
	P_{max}	4
SSA	Leader position update probability	0.5
MVO	WEP_Max	1
	WEP_Min	0.2
GWO	l	[− 1, 1]
	r	[0, 1]
IGWO	l	[− 1, 1]
	r	[0, 1]

Table 4 Twenty-three test functions

Function	Dim	Limits	f_{min}
$F1(c) = \sum_{i=1}^{n} c_i^2$	30	[− 100, 100]	0
$F2(c) = \sum_{i=1}^{n} \lvert c\rvert + \prod_{i}^{n} \lvert c_i \rvert$	30	[− 10, 10]	0
$F3(c) = \sum_{i=1}^{n} (\sum_{j=1}^{i} c_j)^2$	30	[− 100, 100]	0
$F4(c) = max_i \{ \lvert c_i \rvert, \ 1 \le i \le n \}$	30	[− 100, 100]	0
$F5(c) = \sum_{i=1}^{n-1} [100(c_{i+1} - c_i^2)^2 + (c_i - 1)^2]$	30	[− 30, 30]	0
$F6(c) = \sum_{i=1}^{n} ([c_i + 0.5])^2$	30	[− 100, 100]	0
$F7(c) = \sum_{i=1}^{n} ic_i^4 + random[0, 1)$	30	[− 1.28, 1.28]	0
$F8(c) = \sum_{i=1}^{n} -c_i \sin(\sqrt{\lvert c_i \rvert})$	30	[− 500, 500]	− 12569.5
$F9(c) = \sum_{i=1}^{n} [c_i^2 - 10 \cos(2\pi c_i) + 10]$	30	[− 5.12, 5.12]	0
$F10(c) = -20exp(-0.2\sqrt{\frac{1}{n} \sum_{i=1}^{n} c_i^2})$ $-exp(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi c_i)) + 20 + e$	30	[− 32, 32]	0
$F11(c) = \frac{1}{4000} \sum_{i=1}^{n} c_i^2 - \prod_{i=1}^{n} \cos(\frac{c_i}{\sqrt{i}}) + 1$	30	[− 600, 600]	0
$F12(c) = \frac{\pi}{n} 10 \sin \pi y_1)$ $+ \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 sin^2(\pi y_{i+1})] + (y_n - 1)^2$ $+ \sum_{i=1}^{n} u(c_i, 10, 100, 4)$	30	[− 50, 50]	0
$y_i = 1 + \frac{c_i + 1}{4}$			
$u(c_i, a, k, m) = \begin{cases} k(c_i - a)^m & c_i > a \\ 0 & -a < c_i < a \\ k(-c_i - a)^m & c_i < -a \end{cases}$			
$F13(c) = 0.1\{sin^2(3\pi c_1) + \sum_{i=1}^{n} (c_i - 1)^2 [1 + sin^2(3\pi c_i + 1)]$ $+(c_n - 1)^2 [1 + sin^2(2\pi c_n)\} + \sum_{i=1}^{n} u(c_i, 5, 100, 4)$	30	[− 50, 50]	0
$F14(c) = (\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (c_i - a_{ij})^6})^{-1}$	2	[− 65, 65]	1
$F15(c) = \sum_{i=1}^{11} [a_i - \frac{c_1(b_i^2 + b_i c_2)}{b_i^2 + b_i c_3 + c_4}]^2$	4	[− 5, 5]	0.0003
$F16(c) = 4c_1^2 - 2.1c_1^4 + \frac{1}{3} c_1^6 + c_1 c_2 - 4c_2^2 + 4c_2^4$	2	[−5, 5]	−1.0316
$F17(c) = (c_2 - \frac{5.1}{4\pi^2} c_1^2 + \frac{5}{\pi} c_1 - 6)^2 + 10(1 - \frac{1}{8\pi}) \cos c_1 + 10$	2	[− 5, 5]	0.398
$F18(c) = [1 + (c_1 + c_2 + 1)^2 (19 - 14c_1 +$ $3c_1^2 - 14c_2 + 6c_1 c_2 + 3c_2^2)]$ $\times[30 + (2c_1 - 3c_2)^2 (18 - 32c_1 + 12c_1^2 + 48c_2 - 36c_1 c_2 + 27c_2^2)]$	2	[− 2, 2]	3
$F19(c) = -\sum_{i=1}^{4} b_i exp(-\sum_{j=1}^{3} a_{ij}(c_j - p_{ij})^2)$	3	[1, 3]	− 3.86
$F20(c) = -\sum_{i=1}^{4} b_i exp(-\sum_{j=1}^{6} a_{ij}(c_j - p_{ij})^2)$	6	[0, 1]	− 3.32
$F21(c) = -\sum_{i=1}^{5} [(C - a_i)(C - a_i)^T + b_i]^{-1}$	4	[0, 10]	− 10.1532
$F22(c) = -\sum_{i=1}^{7} [(C - a_i)(C - a_i)^T + b_i]^{-1}$	4	[0, 10]	− 10.4028
$F23(c) = -\sum_{i=1}^{10} [(C - a_i)(C - a_i)^T + b_i]^{-1}$	4	[0, 10]	− 10.5363

experiments are performed on Windows 11, Intel Core i3, 2.10GHz, 8.00 GB RAM, MATLAB R2022b (Table 4).

4.1 Sensitivity analysis

The proposed MGWO algorithm employs two parameters, i.e., the number of grey wolves and the maximum number of iterations.

4.2 Number of grey wolves

The MGWO algorithm was simulated for different values of grey wolf (i.e., 10, 15, 20, 25, 30). Figure 4 shows the variations of different numbers of search agents on benchmark test functions. Figure 4 shows that the value of the fitness function reduces as the number of search agents rises.

4.3 Maximum number of iterations

The MGWO algorithm was run for different numbers of iterations. The values of Maximum iteration used in experimentation are 200, 400, 600, 800, and 1000. Figure 5 demonstrates the impact of the number of iterations on benchmark test

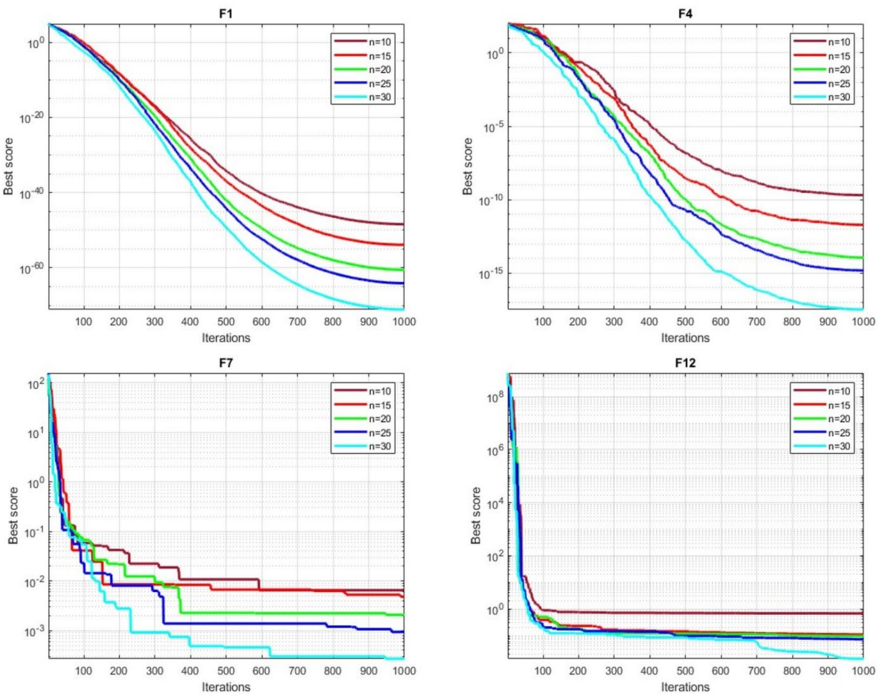


Fig. 4 Sensitivity analysis of the proposed MGWO algorithm for the number of grey wolves

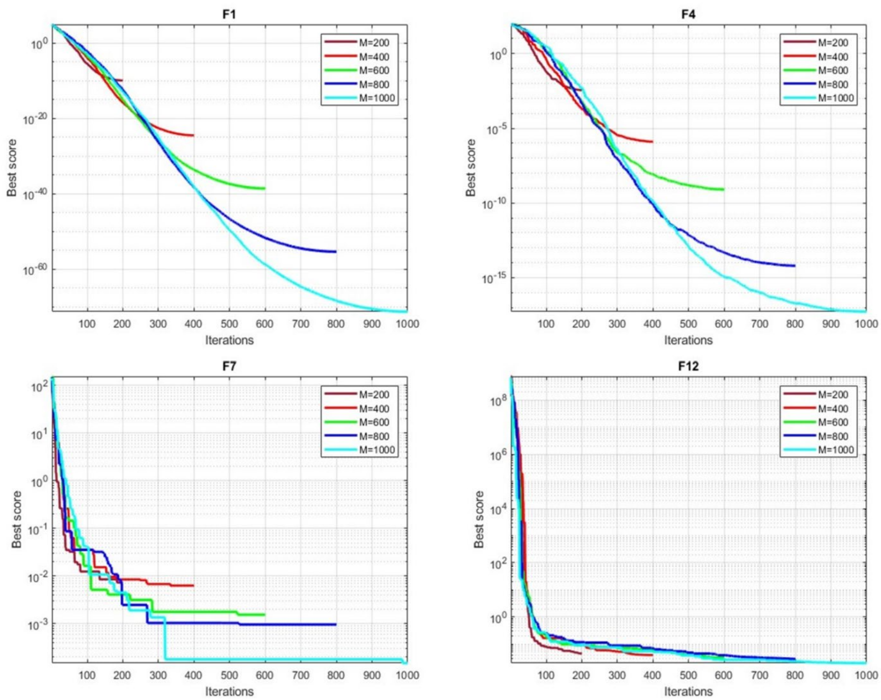


Fig. 5 Sensitivity analysis of the proposed MGWO algorithm for the number of iterations

functions. As the number of iterations increases, the MGWO algorithm converges to the optimum.

4.4 In comparison to other algorithms

To further evaluate the performance of the MGWO, the MGWO algorithm was tested on 23 benchmark functions, and the results were compared with the PSO [15], TSA [20], SSA [121], MVO [39], GWO [16], and IGWO [122] algorithms. Each algorithm is evaluated using the average values and standard deviation after 30 runs, the solution with the highest accuracy is bolded in the table. Tables 5 and 6 displays the results of the 23 test functions. The MGWO's convergence accuracy and optimization capacity can be seen in the average values and standard deviation shown in Tables 5 and 6. When solving the $F1$–$F5$ and $F7$ functions for the seven uni-modal functions, the MGWO performs better in terms of accuracy and standard deviation, even though the optimization accuracy falls short of the theoretically ideal value of 0. When solving the $F9$ and $F11$ functions, the optimization accuracy for the six multi-modal functions reaches the theoretical optimal value of 0, and the algorithm's great robustness and precision of the solution are clearly demonstrated. Meanwhile, when solving the $F8$ and $F10$ functions, the MGWO also yields a better result when compared to other optimization techniques. When it comes to the ten

Table 5 The results of the 23 test functions are mentioned in bold; these results are the global best solutions

Functions	PSO		TSA		SSA		MGWO	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	0.000155	0.000208	6.113E−42	2.22E−41	1.845E−08	4.009E−09	**1.284E−60**	**3.294E−60**
F2	0.041444	0.066589	9.712E−27	7.147E−27	8.541046	23.58543	**2.718E−37**	**3.142E−37**
F3	31.69417	7.391107	3.395E−07	0.000001231	850.2645	560.3619	**1.753E−14**	**6.39E−14**
F4	0.84832	0.231127	0.044329	0.055212	11.47617	4.679967	**9.856E−15**	**1.107E−14**
F5	71.26033	56.07651	28.72735	0.625842	185.6412	312.1713	**27.13666**	**1.06715**
F6	0.000183	0.000249	3.837973	0.583829	**1.894E−08**	**4.241E−09**	1.134553	0.324337
F7	0.092081	0.022263	0.006225	0.001512	0.139053	0.044611	**0.000606**	**0.000395**
F8	−6610.02	709.0768	−5979.17	657.4813	−46074.8	10101.01	**−9941.92**	**548.6993**
F9	56.13828	13.6469	192.6031	58.19319	72.30015	21.43404	**0**	**0**
F10	0.51122	0.582879	0.83678	1.423064	2.318048	0.803747	**8.73E−15**	**2.52E−15**
F11	0.016364	0.017539	0.010615	0.010982	0.009364	0.007439	**0**	**0**
F12	**0.020742**	**0.057099**	9.065166	4.51854	7.53672	4.261936	0.077354	0.031404
F13	**0.00296**	**0.00493**	2.82514	0.412012	11.73447	12.63357	0.961357	0.329458
F14	3.484566	3.485953	7.814166	5.521635	1.19602	0.753589	**0.998**	**4.516E−16**
F15	0.000873	0.000154	0.008181	0.015421	0.001063	0.000111	**0.000309**	**0.0000052**
F16	**−1.0316**	**6.775E−16**	**−1.0316**	**6.775E−16**	**−1.0316**	**6.775E−16**	**−1.0316**	**6.775E−16**
F17	**0.39789**	**1.693E−16**	0.397914	0.000019	**0.39789**	**1.693E−16**	**0.39789**	**1.693E−16**
F18	**3**	**0**	8.4	20.55035	**3**	**0**	**3**	**0**
F19	**−3.8628**	**3.161E−15**	−3.86216	0.001974	**−3.8628**	**3.161E−15**	**−3.8628**	**3.161E−15**
F20	−3.24273	0.057008	−3.15373	0.397481	−3.25371	0.065302	**−3.322**	**1.806E−15**
F21	−7.20113	3.304531	−6.228666	3.637145	−6.978966	3.53603	**−10.1524**	**0.00055**
F22	−8.07678	3.39022	−6.98956	3.542498	−7.974533	3.108462	**−10.4022**	**0.00047**
F23	−8.50334	2.975559	−7.56722	3.539034	−7.292253	3.606672	**−10.5355**	**0.00043**

fixed dimensional multi-modal functions, $F14$, $F15$, and $F20$–$F23$ outperforms other algorithms in terms of value, while the remaining functions' outcomes mostly agree with the contrast algorithm's.

4.5 Statistical analysis

This subsection presents a statistical analysis of the performance of competitor algorithms and MGWO to establish whether or not MGWO has a statistically significant advantage. The Wilcoxon rank sum test [123] is utilized to ascertain the statistically significant difference between the average of two data samples. Using an index defined as a p-value, the Wilcoxon rank sum test is used to establish

Table 6 The results of the 23 test functions are mentioned in bold; these results are the global best solutions

Functions	MVO		GWO		IGWO		MGWO	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	0.536762	0.105211	2.441E−50	4.034E−50	9.507E−52	2.192E−51	**1.284E−60**	**3.294E−60**
F2	0.474956	0.160011	1.076E−29	8.037E−30	2.593E−32	2.252E−32	**2.718E−37**	**3.142E−37**
F3	91.28513	35.4451	9.972E−12	3.368E−11	1.631E−07	4.918E−07	**1.753E−14**	**6.39E−14**
F4	1.366814	0.393004	3.95E−12	3.261E−12	2.994E−09	4.114E−09	**9.856E−15**	**1.107E−14**
F5	177.8109	148.2866	26.94643	0.642726	**23.22303**	**0.218961**	27.13666	1.06715
F6	0.483774	0.132243	0.928531	0.452457	**0.0000011**	**9.553E−07**	1.134553	0.324337
F7	0.024403	0.007886	0.001334	0.000848	0.001984	0.000736	**0.000606**	**0.000395**
F8	−7906.93	644.4043	− 5987.92	737.3029	− 5925.33	785.0471	**−9941.92**	**548.6993**
F9	113.9333	25.94823	0.759486	1.97742	22.58737	8.155451	**0**	**0**
F10	2.376974	4.735421	1.891E−14	5.31E−15	1.607E−14	2.89E−15	**8.73E−15**	**2.52E−15**
F11	0.674638	0.100008	0.005999	0.011864	0.004427	0.008278	**0**	**0**
F12	2.097126	0.888537	0.082644	0.148359	**7.884E−08**	**8.976E−08**	0.077354	0.031404
F13	0.099031	0.058758	0.887274	0.23722	**0.013723**	**0.033489**	0.961357	0.329458
F14	**0.998**	**4.516E−16**	7.182926	4.759544	**0.998**	**4.516E−16**	**0.998**	**4.516E−16**
F15	0.010113	0.016749	0.005721	0.008982	0.000368	0.000232	**0.000309**	**0.0000052**
F16	**− 1.0316**	**6.775E−16**	**− 1.0316**	**6.775E−16**	**− 1.0316**	**6.775E−16**	**− 1.0316**	**6.775E−16**
F17	**0.39789**	**1.693E−16**	0.397898	0.000030	**0.39789**	**1.693E−16**	**0.39789**	**1.693E−16**
F18	**3**	**0**	**3**	**0**	**3**	**0**	**3**	**0**
F19	**− 3.8628**	**3.161E−15**	− 3.86052	0.003228	**− 3.8628**	**3.161E−15**	**− 3.8628**	**3.161E−15**
F20	−3.26598	0.060909	− 3.2534	0.085244	− 3.29822	0.048373	**− 3.322**	**1.806E−15**
F21	−8.46286	2.431177	− 9.81278	1.293253	− 9.1202	2.04485	**−10.1524**	**0.00055**
F22	−7.11098	3.243966	− 10.4025	0.000243	− 10.4029	7.226E−15	**−10.4022**	**0.00047**
F23	−8.91722	2.799761	− 9.99496	2.058642	− 10.5364	9.033E−15	**−10.5355**	**0.00043**

whether or not the statistical advantage of MGWO over any of the competing algorithms is significant. Two-tailed *t*-tests [124] have been used to compare different statistical outcomes at a consequence of 0.05. The *t* values are determined with the help of average and std values. A -*t* value indicates that the statistical outcomes of the MGWO optimization mistakes are significantly less, and vice versa. The corresponding t value is highlighted if the difference is a statistically significant error. The symbols $+/=/-$ represent that MGWO wins functions, ties functions, and loses functions. The statistical outcomes of the optimization mistakes demonstrate that MGWO has much superior total achievement when compared with the other algorithms. Tables 7 and 8 present results of the Wilcoxon rank-sum test and t-test on 23 benchmark functions and validation of the Wilcoxon rank-sum test and t-test on 23 benchmark functions comparing the performance of competing algorithms with MGWO. MGWO outperforms the corresponding algorithm statistically in situations where the p-value and t-value are less than 0.05, according to these data.

4.6 Convergence analysis

Figures 6, 7, and 8 displays the convergence graph of MGWO and other algorithms. As illustrated in Figs. 6, 7, and 8 the suggested method in uni-modal functions adheres to a certain pattern that prioritizes the exploitation stage (functions F1 and F3). The proposed method exhibits a distinct pattern in multi-modal functions with numerous local optimal values. It gives more consideration to the early algorithmic stages of the exploration process. Nevertheless, exploration is carried out in broken form (functions F12 and F13) during the algorithm's final stages, which are often the exploitation phase. The suggested algorithm offers a superior pattern of convergence for almost all functions.

5 MGWO for solving real-life engineering problems

This section evaluates the proposed algorithm performance in three real-life engineering problems using constrained engineering benchmarks. The tension/compression spring, the gear train, and the three-bar truss are all part of the engineering design problems. The MGWO runs independently for each engineering problem 30 times, with a selected grey wolf population size of 30, with 1000 iterations, and a number of function evaluations (NFEs) of 15,000.

5.1 Tension/compression spring design problem

This problem aims to optimize the weight of a tension/compression spring [125], as shown in Fig. 9. The problem has constraints on minimum deflection, shear stress, surge frequency, outside diameter limits, and design variables. The design variables are the mean coil diameter D, the wire diameter d, and the number of active coils N. Table 9 presents the outcomes of this experiment. The MGWO algorithm outperformed other algorithms in this problem.

Table 7 Results of Wilcoxon rank-sum test and t-test on 23 benchmark functions

Functions	MGWO versus PSO		MGWO versus TSA		MGWO versus SSA		MGWO versus MVO		MGWO versus GWO		MGWO versus IGWO	
	p-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test
F1	1.8E−06	− 4.1E+00	1.8E−06	− 1.5E+00	1.8E−06	− 2.5E+01	1.8E−06	− 2.8E+01	1.8E−06	− 3.3E+00	1.8E−06	− 2.4E+00
F2	1.8E−06	− 3.4E+00	1.8E−06	− 7.4E+00	1.8E−06	− 2.0E+00	1.8E−06	− 1.6E+01	1.8E−06	− 7.3E+00	1.8E−06	− 6.3E+00
F3	1.8E−06	− 2.3E+01	1.8E−06	− 1.5E+00	1.8E−06	− 8.3E+00	1.8E−06	− 1.4E+01	1.1E−05	− 1.6E+00	1.8E−06	− 1.8E+00
F4	1.8E−06	− 2.0E+01	1.8E−06	− 4.4E+00	1.8E−06	− 1.3E+01	1.8E−06	− 1.9E+01	1.8E−06	− 6.6E+00	1.8E−06	− 4.0E+00
F5	6.4E−05	− 4.3E+00	2.4E−06	− 7.0E+00	1.5E−04	− 2.8E+00	1.8E−06	− 5.6E+00	7.7E−01	8.4E−01	1.8E−06	2.0E+01
F6	1.8E−06	1.9E+01	1.8E−06	− 2.2E+01	1.8E−06	1.9E+01	1.8E−06	1.0E+01	7.6E−02	2.0E+00	1.8E−06	1.9E+01
F7	1.8E−06	− 2.3E+01	1.8E−06	− 2.0E+01	1.8E−06	− 1.7E+01	1.8E−06	− 1.7E+01	5.4E−04	− 4.3E+00	1.8E−06	− 9.0E+00
F8	1.1E−02	3.5E+00	7.5E−01	2.9E−01	1.8E−06	2.2E+01	1.8E−06	1.1E+01	8.2E−01	3.2E−01	1.8E−06	2.3E+01
F9	1.8E−06	− 2.3E+01	1.8E−06	− 1.8E+01	1.7E−06	− 1.8E+01	1.8E−06	− 2.4E+01	1.2E−04	− 2.1E+00	1.8E−06	− 1.5E+01
F10	1.8E−06	− 4.8E+00	7.4E−06	− 3.2E+00	1.8E−06	− 1.6E+01	1.8E−06	− 2.7E+00	3.1E−06	− 9.5E+00	1.4E−06	− 1.0E+01
F11	1.8E−06	− 5.1E+00	2.0E−04	− 5.3E+00	1.8E−06	− 6.9E+00	1.8E−06	− 3.7E+01	7.8E−03	− 2.8E+00	7.8E−03	− 2.9E+00
F12	1.1E−03	4.8E+00	1.8E−06	− 1.1E+01	1.8E−06	− 9.6E+00	1.8E−06	− 1.2E+01	4.5E−02	− 1.9E−01	1.8E−06	1.3E+01
F13	1.8E−06	1.6E+01	1.8E−06	− 1.9E+01	2.7E−04	− 4.7E+00	1.8E−06	1.4E+01	3.1E−01	1.0E+00	1.8E−06	1.6E+01
F14	7.1E−05	− 3.9E+00	6.6E−06	− 6.8E+00	NA	− 1.4E+00	NA	0.0E+00	6.7E−06	− 7.1E+00	NA	0.0E+00
F15	1.8E−06	− 2.0E+01	1.8E−06	− 2.8E+00	1.8E−06	− 3.7E+01	1.8E−06	− 3.2E+00	5.3E−04	− 3.3E+00	3.4E−04	− 1.4E+00
F16	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00
F17	NA	0.0E+00	1.7E−05	− 6.8E+00	NA	0.0E+00	NA	0.0E+00	NA	− 1.4E+00	NA	0.0E+00
F18	NA	0.0E+00	NA	− 2.2E+00	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00	NA	0.0E+00
F19	NA	0.0E+00	1.4E−06	− 1.8E+00	NA	0.0E+00	NA	0.0E+00	1.9E−04	− 3.9E+00	NA	0.0E+00
F20	8.1E−06	− 7.6E+00	1.8E−06	− 2.3E+00	4.5E−04	− 5.7E+00	1.2E−04	− 5.0E+00	1.2E−04	− 4.4E+00	3.1E−02	− 2.7E+00
F21	4.8E−02	− 4.9E+00	1.8E−06	− 5.9E+00	4.8E−02	− 4.9E+00	3.2E−01	− 3.8E+00	5.4E−01	− 1.4E+00	6.5E−01	− 2.8E+00
F22	3.2E−01	− 3.8E+00	1.8E−06	− 5.3E+00	1.3E−01	− 4.3E+00	8.8E−03	− 5.6E+00	2.6E−02	2.9E+00	8.3E−06	7.9E+00
F23	6.5E−01	− 3.7E+00	1.8E−06	− 4.6E+00	4.8E−02	− 4.9E+00	6.8E−01	− 3.2E+00	5.3E−02	− 1.4E+00	1.6E−06	1.1E+01

Table 8 Validation of Wilcoxon rank-sum test and t-test on 23 benchmark functions

Functions	MGWO versus PSO		MGWO versus TSA		MGWO versus SSA		MGWO versus MVO		MGWO versus GWO		MGWO versus IGWO	
	p-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test
F1	√	+	√	+	√	+	√	+	√	+	√	+
F2	√	+	√	+	√	+	√	+	√	+	√	+
F3	√	+	√	+	√	+	√	+	√	+	√	+
F4	√	+	√	+	√	+	√	+	√	+	√	+
F5	√	+	√	+	√	+	√	+	×	=	√	−
F6	√	−	√	+	√	−	√	−	×	−	√	−
F7	√	+	√	+	√	+	√	+	√	+	√	+
F8	×	−	×	=	√	−	√	−	×	=	√	−
F9	√	+	√	+	√	+	√	+	√	+	√	+
F10	√	+	√	+	√	+	√	+	√	+	√	+
F11	√	+	√	+	√	+	√	+	√	+	√	+
F12	√	−	√	+	√	+	√	+	√	=	√	−
F13	√	−	√	+	√	+	√	−	×	=	√	−
F14	√	+	√	+	√	+	√	=	√	+	√	=
F15	√	+	√	+	√	+	√	+	√	+	√	+
F16	√	=	√	=	√	=	√	=	√	=	√	=
F17	√	=	√	+	√	=	√	=	√	+	√	=
F18	√	=	√	+	√	=	√	=	√	=	√	=
F19	√	=	√	+	√	=	√	=	√	+	√	=
F20	√	+	√	+	√	+	√	+	√	+	√	+

Table 8 (continued)

Functions	MGWO versus PSO		MGWO versus TSA		MGWO versus SSA		MGWO versus MVO		MGWO versus GWO		MGWO versus IGWO	
	p-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test	*p*-value	*t*-test
F21	√	+	√	+	√	+	×	+	×	+	×	+
F22	×	+	√	+	×	+	√	+	√	−	√	−
F23	×	+	√	+	√	+	×	+	×	+	√	−

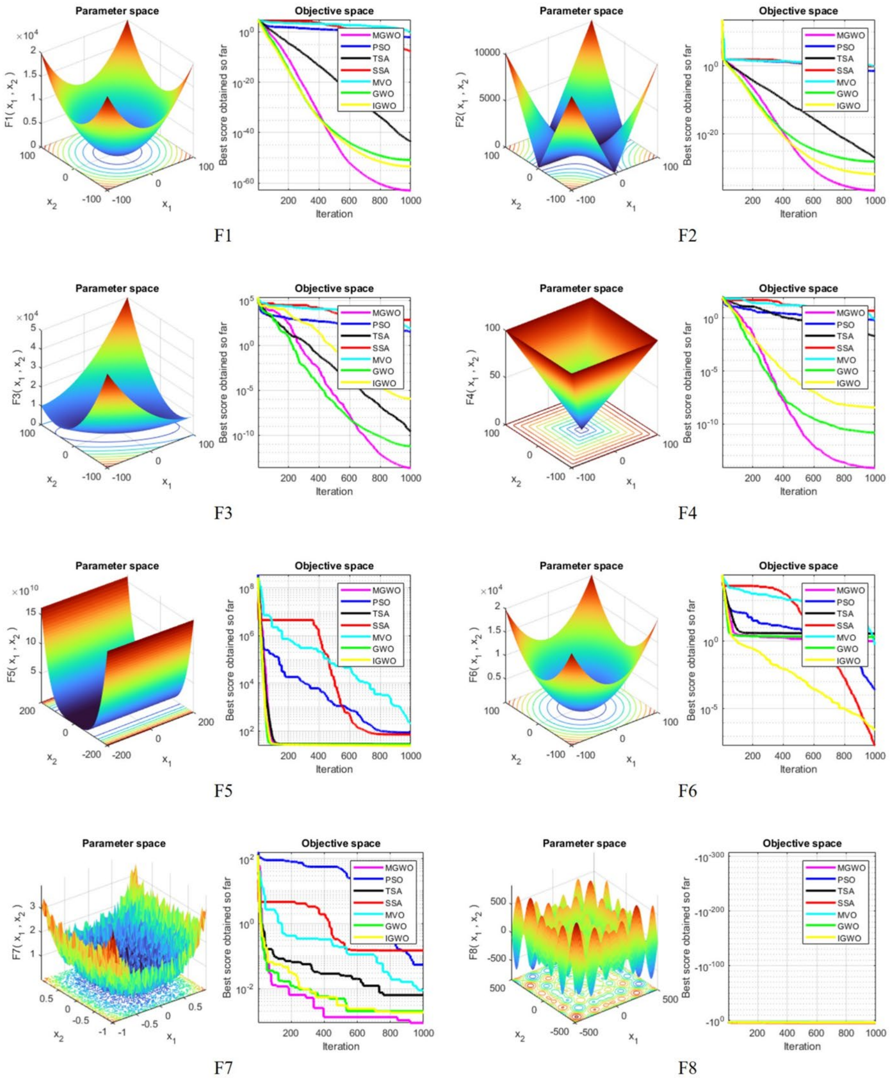


Fig. 6 Convergence graph of MGWO and other algorithms

5.2 Gear train design problem

Sandgren presented the gear train design problem [126, 127], an unconstrained discrete problem in mechanical engineering. This benchmark task aims to minimize the gear ratio, which is the ratio of the angular velocity of the output shaft to the input shaft. The number of teeth of gears $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{C}_3$, and $\mathcal{C}_4$ are considered as the design variables, as shown in Fig. 10. Table 10 presents the outcomes of this experiment. The MGWO algorithm outperformed other algorithms in this problem.

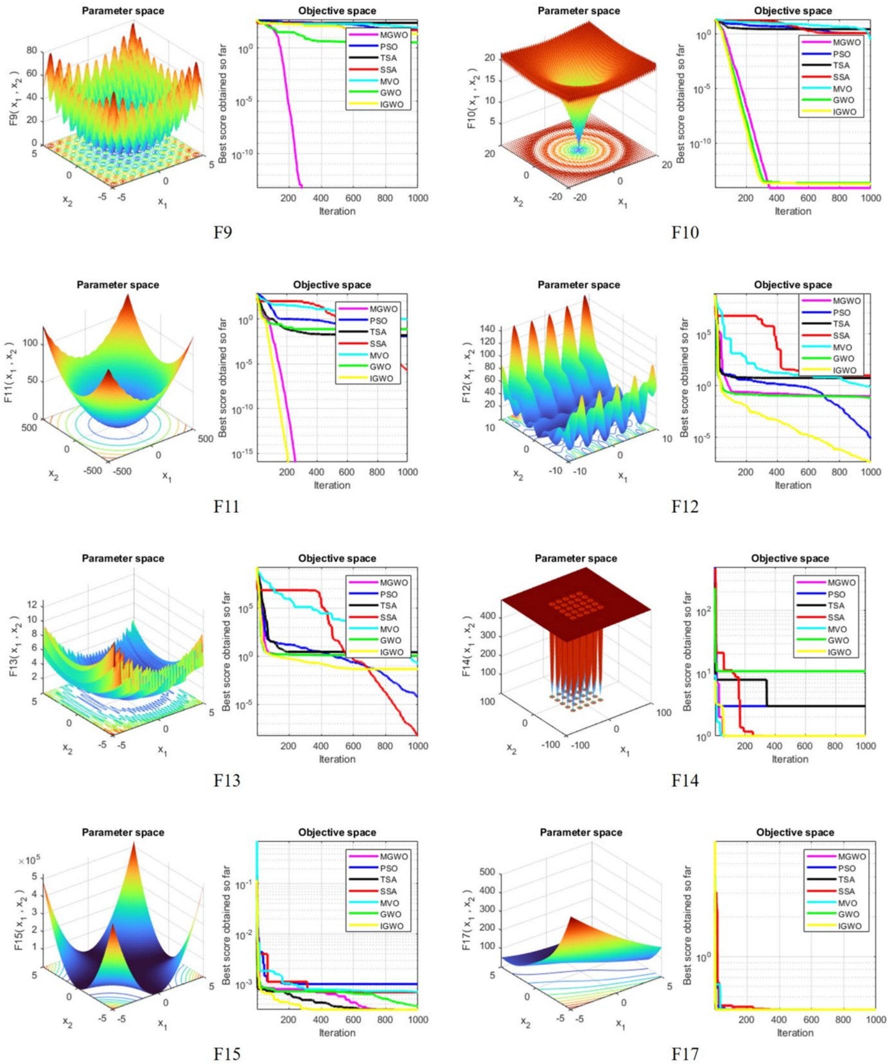


Fig. 7 Convergence graph of MGWO and other algorithms

5.3 Three-bar truss design problem

This optimization problem from civil engineering has a confined and troublesome space [128]. The primary goal of this challenge is to reduce the weight of bar constructions. The restrictions for this problem are determined by the stress constraints of each bar. The resulting problem contains a non-linear objective function and three non-linear constraints, as shown in Fig. 11. The results are presented in Table 11 The proposed method successfully identified the optimal value for the problem.

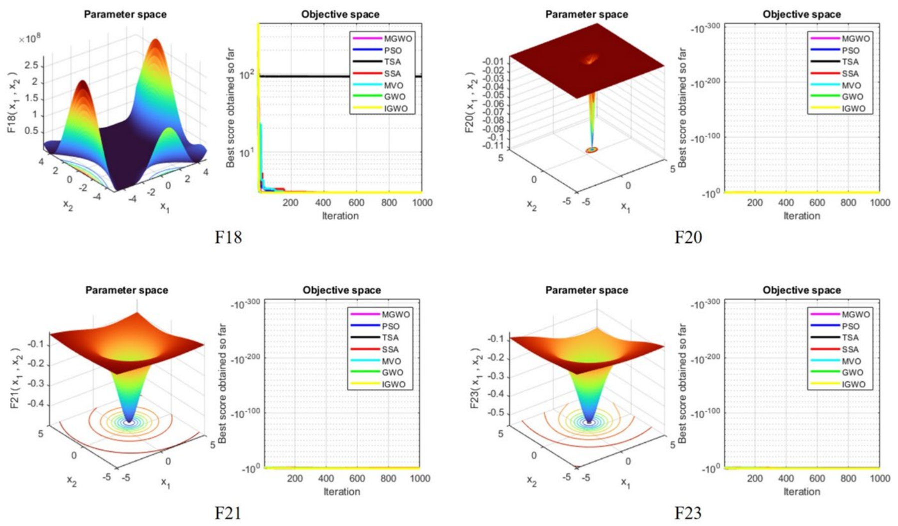


Fig. 8 Convergence graph of MGWO and other algorithms

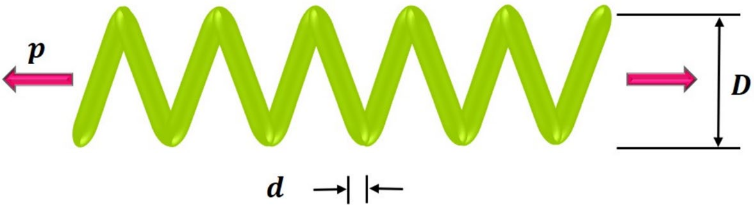


Fig. 9 The design of the tension/compression spring problem

Table 9 The comparison outcomes of the tension/compression spring problem

Algorithms	Optimal values for variables			Optimum weight
	D	*d*	*N*	
MGWO	0.051437039	0.350447807	11.68029942	**0.012684**
TSA	0.057637718	0.517253228	5.72736214	0.013278
SSA	0.13678061	1.432553267	1.089018486	7.63E+14
MVO	0.069209835	0.941963233	2.026422819	0.018167
GWO	0.05	0.317347426	14.04051804	0.012726
IGWO	0.050388413	0.326167535	13.34326875	0.012706

Best result highlighted in bold

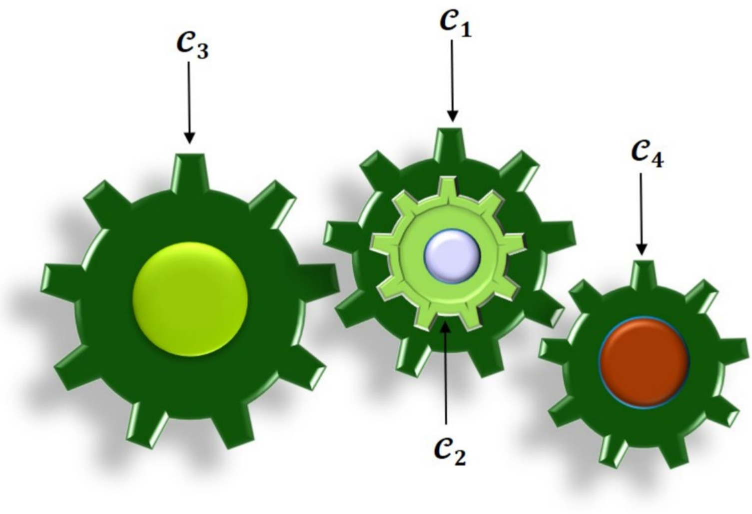


Fig. 10 The design of the gear train problem

Table 10 The comparison outcomes of the gear train problem

Algorithms	Optimal values for variables				Optimum weight
	c_1	c_2	c_3	c_4	
MGWO	43.23451442	19.48161573	16.80898646	52.49664159	**9.444E−15**
TSA	41.38908813	14.89380456	22.81969856	56.91350992	1.153E−11
SSA	50.80890822	14.00937474	29.37997485	42.34395302	0.0022119
MVO	30.42669549	15.51261593	12	42.40446052	1.896E−12
GWO	60	30.57637562	15.04813329	53.1507917	1.386E−12
IGWO	59.38921756	13.10291292	12.58277378	19.24141193	2.135E−12

Best result highlighted in bold

6 Conclusion

The original GWO algorithm has premature convergence and poor accuracy while solving global optimization problems. In this study, a modified GWO is proposed to overcome the shortcomings. The MGWO algorithm is proposed by modifying the position update equation of the original GWO algorithm. We investigated 23 functions with various features, including uni-modal, multi-modal, and fixed-dimensional multi-modal, and compared the outcomes to six algorithms. The experimental results indicate that the MGWO algorithm outperforms compare with six different algorithms in terms of optimization performance and stability. Then three real-life engineering optimization design problems (tension/compression spring, gear train, and three-bar truss) are solved using various objective functions,

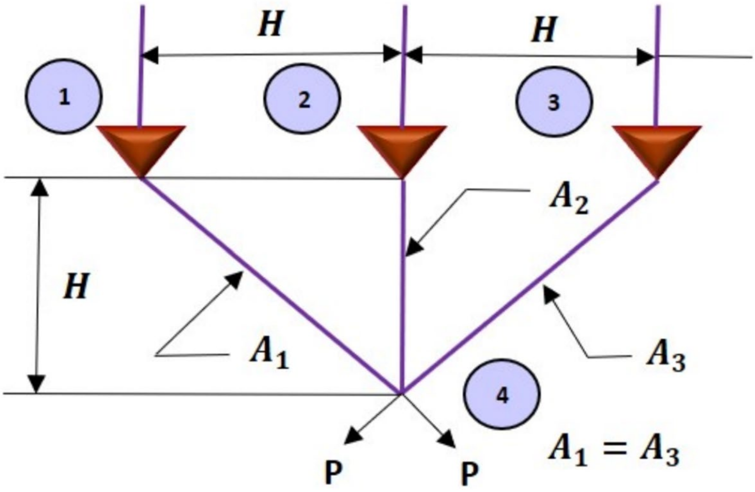


Fig. 11 The design of the three-bar truss problem

Table 11 The comparison outcomes of the three-bar truss problem

Algorithms	Optimal values for variables		Optimum weight
	A_1	A_2	
MGWO	0.78930035	0.40648617	**263.8964694**
TSA	0.79533361	0.38978063	263.9323795
SSA	0.73708847	0.63847416	272.3275195
MVO	0.78649751	0.41444474	263.8995652
GWO	0.78473988	0.41952331	263.9102876
IGWO	0.78998884	0.40457536	263.9001244

Best result highlighted in bold

constraint conditions, and features. Meanwhile, the Wilcoxon rank-sum test and t-test were used to evaluate the results of the MGWO algorithm. The experimental results demonstrate that the MGWO algorithm outperforms other comparison algorithms and is capable of dealing with engineering design problems. However, the proposed MGWO algorithm has shown insignificant and mediocre results for one uni-modal (F6) and two multi-modal (F12 and F13) functions. In future work, the MGWO suggests several improvements, such as the inclusion of adaptive inertia factors, image segmentation, feature selection, levy flight distribution, binary, and multi-objective problems.

Acknowledgements The author would like to thanks VIT University for supporting this research work.

Author contributions Gopi S: Conceptualization, Methodology, Writing-original draft.
Prabhujit Mohapatra: Conceptualization, Methodology, Supervision, Writing-review and editing.

Funding No funding was received for conducting this study.

Data availability All data generated or analyzed during this study are included in this article.

Declarations

Conflict of interest The authors declare that they have no confict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Dhiman, G.: SSC: a hybrid nature-inspired meta-heuristic optimization algorithm for engineering applications. Knowl. Based Syst. **222**, 106926 (2021)
2. Dehghani, M., Hubálovskỳ, Š, Trojovskỳ, P.: Tasmanian devil optimization: a new bio-inspired optimization algorithm for solving optimization algorithm. IEEE Access **10**, 19599–19620 (2022)
3. Mohapatra, P., Das, K.N., Roy, S.: A modified competitive swarm optimizer for large scale optimization problems. Appl. Soft Comput. **59**, 340–362 (2017)
4. Chen, X., Mei, C., Xu, B., Yu, K., Huang, X.: Quadratic interpolation based teaching-learning-based optimization for chemical dynamic system optimization. Knowl. Based Syst. **145**, 250–263 (2018)
5. Mohapatra, P., Roy, S., Das, K.N., Dutta, S., Raju, M.S.S.: A review of evolutionary algorithms in solving large scale benchmark optimisation problems. Int. J. Math. Oper. Res. **21**(1), 104–126 (2022)
6. Francisco, M., Revollar, S., Vega, P., Lamanna, R.: A comparative study of deterministic and stochastic optimization methods for integrated design of processes. IFAC Proc. Vol. **38**(1), 335–340 (2005)
7. Storn, R., Price, K.: Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**, 341–359 (1997)
8. Whitley, D.: A genetic algorithm tutorial. Stat. Comput. **4**, 65–85 (1994)
9. Espejo, P.G., Ventura, S., Herrera, F.: A survey on the application of genetic programming to classification. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **40**(2), 121–144 (2009)
10. Beyer, H.-G., Schwefel, H.-P.: Evolution strategies: a comprehensive introduction. Nat. Comput. **1**, 3–52 (2002)
11. Reynolds, R.G., Peng, B.: Cultural algorithms: computational modeling of how cultures learn to solve problems—an engineering example. Cybern. Syst. Int. J. **36**(8), 753–771 (2005)
12. Castro, Ld., Timmis, J.I.: Artificial immune systems as a novel soft computing paradigm. Soft. Comput. **7**, 526–544 (2003)
13. Hofmeyr, S.A., Forrest, S.: Architecture for an artificial immune system. Evolut. Comput. **8**(4), 443–473 (2000)
14. Simon, D.: Biogeography-based optimization. IEEE Trans. Evolut. Comput. **12**(6), 702–713 (2008)
15. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
16. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
17. Dorigo, M., Birattari, M., Stutzle, T.: Ant colony optimization. IEEE Comput. Intell. Mag. **1**(4), 28–39 (2006)
18. Faramarzi, A., Heidarinejad, M., Mirjalili, S., Gandomi, A.H.: Marine predators algorithm: a nature-inspired metaheuristic. Expert Syst. Appl. **152**, 113377 (2020)
19. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. **95**, 51–67 (2016)
20. Kaur, S., Awasthi, L.K., Sangal, A., Dhiman, G.: Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng. Appl. Artif. Intell. **90**, 103541 (2020)
21. Braik, M., Hammouri, A., Atwan, J., Al-Betar, M.A., Awadallah, M.A.: White shark optimizer: a novel bio-inspired meta-heuristic algorithm for global optimization problems. Knowl. Based Syst. **243**, 108457 (2022)
22. MiarNaeimi, F., Azizyan, G., Rashki, M.: Horse herd optimization algorithm: a nature-inspired algorithm for high-dimensional optimization problems. Knowl. Based Syst. **213**, 106711 (2021)

23. Hashim, F.A., Hussien, A.G.: Snake optimizer: a novel meta-heuristic optimization algorithm. Knowl. Based Syst. **242**, 108320 (2022)
24. Li, S., Chen, H., Wang, M., Heidari, A.A., Mirjalili, S.: Slime mould algorithm: a new method for stochastic optimization. Future Gener. Comput. Syst. **111**, 300–323 (2020)
25. Wang, G.-G., Deb, S., Cui, Z.: Monarch butterfly optimization. Neural Comput. Appl. **31**, 1995–2014 (2019)
26. Tu, J., Chen, H., Wang, M., Gandomi, A.H.: The colony predation algorithm. J. Bionic Eng. **18**, 674–710 (2021)
27. Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., Chen, H.: Harris hawks optimization: algorithm and applications. Future Gener. Comput. Syst. **97**, 849–872 (2019)
28. Wang, G.-G.: Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. Memet. Comput. **10**(2), 151–164 (2018)
29. Ahmadianfar, I., Heidari, A.A., Gandomi, A.H., Chu, X., Chen, H.: RUN beyond the metaphor: an efficient optimization algorithm based on Runge Kutta method. Expert Syst. Appl. **181**, 115079 (2021)
30. Dhiman, G., Kumar, V.: Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. Knowl. Based Syst. **159**, 20–50 (2018)
31. Jiang, Y., Wu, Q., Zhu, S., Zhang, L.: Orca predation algorithm: a novel bio-inspired algorithm for global optimization problems. Expert Syst. Appl. **188**, 116026 (2022)
32. Zhao, W., Wang, L., Mirjalili, S.: Artificial hummingbird algorithm: a new bio-inspired optimizer with its engineering applications. Comput. Methods Appl. Mech. Eng. **388**, 114194 (2022)
33. Braik, M.S.: Chameleon swarm algorithm: a bio-inspired optimizer for solving engineering design problems. Expert Syst. Appl. **174**, 114685 (2021)
34. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W., Gandomi, A.H.: Reptile search algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert Syst. Appl. **191**, 116158 (2022)
35. Bertsimas, D., Tsitsiklis, J.: Simulated annealing. Stat. Sci. **8**(1), 10–15 (1993)
36. Dehghani, M., Montazeri, Z., Dhiman, G., Malik, O., Morales-Menendez, R., Ramirez-Mendoza, R.A., Dehghani, A., Guerrero, J.M., Parra-Arroyo, L.: A spring search algorithm applied to engineering optimization problems. Appl. Sci. **10**(18), 6173 (2020)
37. Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)
38. Dehghani, M., Samet, H.: Momentum search algorithm: a new meta-heuristic optimization algorithm inspired by momentum conservation law. SN Appl. Sci. **2**(10), 1720 (2020)
39. Mirjalili, S., Mirjalili, S.M., Hatamlou, A.: Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput. Appl. **27**, 495–513 (2016)
40. Faramarzi, A., Heidarinejad, M., Stephens, B., Mirjalili, S.: Equilibrium optimizer: a novel optimization algorithm. Knowl. Based Syst. **191**, 105190 (2020)
41. Eskandar, H., Sadollah, A., Bahreininejad, A., Hamdi, M.: Water cycle algorithm: a novel metaheuristic optimization method for solving constrained engineering optimization problems. Comput. Struct. **110**, 151–166 (2012)
42. Hashim, F.A., Hussain, K., Houssein, E.H., Mabrouk, M.S., Al-Atabany, W.: Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. Appl. Intell. **51**, 1531–1551 (2021)
43. Hashim, F.A., Houssein, E.H., Mabrouk, M.S., Al-Atabany, W., Mirjalili, S.: Henry gas solubility optimization: a novel physics-based algorithm. Future Gener. Comput. Syst. **101**, 646–667 (2019)
44. Ahmadianfar, I., Heidari, A.A., Noshadian, S., Chen, H., Gandomi, A.H.: Info: an efficient optimization algorithm based on weighted mean of vectors. Expert Syst. Appl. **195**, 116516 (2022)
45. Kaveh, A., Dadras, A.: A novel meta-heuristic optimization algorithm: thermal exchange optimization. Adv. Eng. Softw. **110**, 69–84 (2017)
46. Rao, R.V., Savsani, V.J., Vakharia, D.: Teaching-learning-based optimization: an optimization method for continuous non-linear large scale problems. Inf. Sci. **183**(1), 1–15 (2012)
47. Dehghani, M., Mardaneh, M., Malik, O.: FOA:'following' optimization algorithm for solving power engineering optimization problems. J. Oper. Autom. Power Eng. **8**(1), 57–64 (2020)
48. Atashpaz-Gargari, E., Lucas, C.: Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: 2007 IEEE Congress on Evolutionary Computation, pp. 4661–4667. IEEE (2007)
49. Moosavi, S.H.S., Bardsiri, V.K.: Poor and rich optimization algorithm: a new human-based and multi populations algorithm. Eng. Appl. Artif. Intell. **86**, 165–181 (2019)

50. Zhang, Y., Jin, Z.: Group teaching optimization algorithm: a novel metaheuristic method for solving global optimization problems. Expert Syst. Appl. **148**, 113246 (2020)
51. Kashan, A.H.: An efficient algorithm for constrained global optimization and application to mechanical engineering design: league championship algorithm (LCA). Comput. Aided Des. **43**(12), 1769–1792 (2011)
52. Yang, Y., Chen, H., Heidari, A.A., Gandomi, A.H.: Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. Expert Syst. Appl. **177**, 114864 (2021)
53. Askari, Q., Younas, I., Saeed, M.: Political optimizer: a novel socio-inspired meta-heuristic for global optimization. Knowl. Based Syst. **195**, 105709 (2020)
54. Shi, Y.: Brain storm optimization algorithm. In: Advances in Swarm Intelligence: Second International Conference, ICSI 2011, Chongqing, China, June 12–15, 2011, Proceedings, Part I 2, pp. 303–309. Springer (2011)
55. Wang, C., Zhang, X., Niu, Y., Gao, S., Jiang, J., Zhang, Z., Yu, P., Dong, H.: Dual-population social group optimization algorithm based on human social group behavior law. IEEE Trans. Comput. Soc. Syst. **10**(1), 166–177 (2022)
56. Panwar, D., Saini, G., Agarwal, P.: Human eye vision algorithm (HEVA): a novel approach for the optimization of combinatorial problems. In: Artificial Intelligence in Healthcare, pp. 61–71 (2022)
57. Mousavirad, S.J., Ebrahimpour-Komleh, H.: Human mental search: a new population-based metaheuristic optimization algorithm. Appl. Intell. **47**, 850–887 (2017)
58. Gopi, S., Mohapatra, P.: A modified whale optimisation algorithm to solve global optimisation problems. In: Proceedings of 7th International Conference on Harmony Search, Soft Computing and Applications: ICHSA 2022, pp. 465–477. Springer (2022)
59. Hussain, K., Neggaz, N., Zhu, W., Houssein, E.H.: An efficient hybrid sine-cosine Harris hawks optimization for low and high-dimensional feature selection. Expert Syst. Appl. **176**, 114778 (2021)
60. Gopi, S., Mohapatra, P.: Opposition-based learning cooking algorithm (OLCA) for solving global optimization and engineering problems. Int. J. Mod. Phys. C **35**, 1–28 (2023)
61. Mafarja, M.M., Mirjalili, S.: Hybrid binary ant lion optimizer with rough set and approximate entropy reducts for feature selection. Soft. Comput. **23**(15), 6249–6265 (2019)
62. Sarangi, P., Mohapatra, P.: Evolved opposition-based mountain gazelle optimizer to solve optimization problems. J. King Saud Univ. Comput. Inf. Sci. **35**(10), 101812 (2023)
63. Cheng, Z., Song, H., Wang, J., Zhang, H., Chang, T., Zhang, M.: Hybrid firefly algorithm with grouping attraction for constrained optimization problem. Knowl. Based Syst. **220**, 106937 (2021)
64. Mohapatra, S., Sarangi, P., Mohapatra, P.: An improvised grey wolf optimiser for global optimisation problems. Int. J. Math. Oper. Res. **26**(2), 263–281 (2023)
65. Sarangi, P., Mohapatra, P.: Modified hybrid GWO-SCA algorithm for solving optimization problems. In: International Conference on Data Analytics and Computing, pp. 121–128. Springer (2022)
66. Pelusi, D., Mascella, R., Tallini, L., Nayak, J., Naik, B., Deng, Y.: An improved moth-flame optimization algorithm with hybrid search phase. Knowl. Based Syst. **191**, 105277 (2020)
67. Qin, A.K., Suganthan, P.N.: Self-adaptive differential evolution algorithm for numerical optimization. In: 2005 IEEE Congress on Evolutionary Computation, vol. 2, pp. 1785–1791. IEEE (2005)
68. Chandran, V., Mohapatra, P.: Enhanced opposition-based grey wolf optimizer for global optimization and engineering design problems. Alex. Eng. J. **76**, 429–467 (2023)
69. Gopi, S., Mohapatra, P.: Fast random opposition-based learning Aquila optimization algorithm. Heliyon **10**(4), 26187–26187 (2024)
70. Shankar, K., Eswaran, P.: A secure visual secret share (VSS) creation scheme in visual cryptography using elliptic curve cryptography with optimization technique. Aust. J. Basic Appl. Sci. **9**(36), 150–163 (2015)
71. Yusof, Y., Mustaffa, Z.: Time series forecasting of energy commodity using grey wolf optimizer (2015)
72. Emary, E., Zawbaa, H.M., Grosan, C., Hassenian, A.E.: Feature subset selection approach by gray-wolf optimization. In: Afro-European Conference for Industrial Advancement: Proceedings of the First International Afro-European Conference for Industrial Advancement AECIA 2014, pp. 1–13. Springer (2015)

73. Kamboj, V.K., Bath, S., Dhillon, J.: Solution of non-convex economic load dispatch problem using grey wolf optimizer. Neural Comput. Appl. **27**, 1301–1316 (2016)
74. El-Fergany, A.A., Hasanien, H.M.: Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. Electric Power Compon. Syst. **43**(13), 1548–1559 (2015)
75. Gholizadeh, S.: Optimal design of double layer grids considering nonlinear behaviour by sequential grey wolf algorithm. J. Optim. Civ. Eng. **5**(4), 511–523 (2015)
76. Komaki, G., Kayvanfar, V.: Grey wolf optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time. J. Comput. Sci. **8**, 109–120 (2015)
77. Emary, E., Zawbaa, H.M., Hassanien, A.E.: Binary grey wolf optimization approaches for feature selection. Neurocomputing **172**, 371–381 (2016)
78. Pan, T.-S., Dao, T.-K., Nguyen, T.-T., Chu, S.-C.: A communication strategy for paralleling grey wolf optimizer. In: Genetic and Evolutionary Computing: Proceedings of the Ninth International Conference on Genetic and Evolutionary Computing, August 26–28, 2015, Yangon, Myanmar-Volume II 9, pp. 253–262. Springer (2016)
79. Jayapriya, J., Arock, M.: A parallel GWO technique for aligning multiple molecular sequences. In: 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 210–215. IEEE (2015)
80. Jitkongchuen, D.: A hybrid differential evolution with grey wolf optimizer for continuous global optimization. In: 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 51–54. IEEE (2015)
81. Tawhid, M.A., Ali, A.F.: A hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function. Memet. Comput. **9**, 347–359 (2017)
82. Zhang, S., Luo, Q., Zhou, Y.: Hybrid grey wolf optimizer using elite opposition-based learning strategy and simplex method. Int. J. Comput. Intell. Appl. **16**(02), 1750012 (2017)
83. Singh, N., Singh, S.: A modified mean gray wolf optimization approach for benchmark and biomedical problems. Evolut. Bioinform. **13**, 1176934317729413 (2017)
84. Zhu, A., Xu, C., Li, Z., Wu, J., Liu, Z.: 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 (2015)
85. Singh, N., Singh, S., et al.: Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. J. Appl. Math. **2017**, 2030489 (2017)
86. Kumar, P.S.: The PSK method: a new and efficient approach to solving fuzzy transportation problems. In: Transport and Logistics Planning and Optimization, pp. 149–197 (2023)
87. Kumar, P.S.: The theory and applications of the software-based PSK method for solving intuitionistic fuzzy solid transportation problems. In: Perspectives and Considerations on the Evolution of Smart Systems, pp. 137–186 (2023)
88. Kumar, P.S.: Algorithms for solving the optimization problems using fuzzy and intuitionistic fuzzy set. Int. J. Syst. Assur. Eng. Manag. **11**(1), 189–222 (2020)
89. Kumar, P.S.: Developing a new approach to solve solid assignment problems under intuitionistic fuzzy environment. Int. J. Fuzzy Syst. Appl. (IJFSA) **9**(1), 1–34 (2020)
90. Kumar, P.S.: Intuitionistic fuzzy solid assignment problems: a software-based approach. Int. J. Syst. Assur. Eng. Manag. **10**(4), 661–675 (2019)
91. Kumar, P.S.: Computationally simple and efficient method for solving real-life mixed intuitionistic fuzzy 3D assignment problems. Int. J. Softw. Sci. Comput. Intell. (IJSSCI) **14**(1), 1–42 (2022)
92. Friedman, J.H.: Multivariate adaptive regression splines. Ann. Stat. **19**(1), 1–67 (1991)
93. Özmen, A., Weber, G.-W., Kropat, E.: Robustification of conic generalized partial linear models under polyhedral uncertainty. Methods **20**, 21–22 (2012)
94. Özmen, A., Zinchenko, Y., Weber, G.-W.: Robust multivariate adaptive regression splines under cross-polytope uncertainty: an application in a natural gas market. Ann. Oper. Res. **324**(1), 1337–1367 (2023)
95. Özmen, A., Kropat, E., Weber, G.-W.: Robust optimization in spline regression models for multi-model regulatory networks under polyhedral uncertainty. Optimization **66**(12), 2135–2155 (2017)
96. Kropat, E., Özmen, A., Weber, G.-W., Meyer-Nieberg, S., Defterli, O.: Fuzzy prediction strategies for gene-environment networks-fuzzy regression analysis for two-modal regulatory systems. RAIRO Oper. Res. Rech. Opér. **50**(2), 413–435 (2016)

97. Kropat, E., Meyer-Nieberg, S.: A multi-layered adaptive network approach for shortest path planning during critical operations in dynamically changing and uncertain environments. In: 2016 49th Hawaii International Conference on System Sciences (HICSS), pp. 1369–1378. IEEE (2016)
98. Özmen, A., Kropat, E., Weber, G.-W.: Spline regression models for complex multi-modal regulatory networks. Optim. Methods Softw. **29**(3), 515–534 (2014)
99. Weber, G.W., Kropat, E., Tezel, A., Belen, S.: Optimization applied on regulatory and eco-finance networks-survey and new developments (2010)
100. Kropat, E., Tikidji-Hamburyan, R.A., Weber, G.-W.: Operations research in neuroscience. Ann. Oper. Res. **258**, 1–4 (2017)
101. Weber, G.-W., Batmaz, I., Köksal, G., Taylan, P., Yerlikaya-Özkurt, F.: CMARS: a new contribution to nonparametric regression with multivariate adaptive regression splines supported by continuous optimization. Inverse Probl. Sci. Eng. **20**(3), 371–400 (2012)
102. Taylan, P., Weber, G.-W., Yerlikaya Özkurt, F.: A new approach to multivariate adaptive regression splines by using Tikhonov regularization and continuous optimization. TOP **18**, 377–395 (2010)
103. Kalaycı, B., Purutçuoğlu, V., Weber, G.W.: Operation research in neuroscience: a recent perspective of operation research application in finance. In: Operations Research: New Paradigms and Emerging Applications, pp. 170–190 (2022)
104. Kalaycı, B., Özmen, A., Weber, G.-W.: Mutual relevance of investor sentiment and finance by modeling coupled stochastic systems with MARS. Ann. Oper. Res. **295**, 183–206 (2020)
105. Graczyk-Kucharska, M., Szafrański, M., Gütmen, S., Çevik, A., Weber, G.-W., Włodarczyk, Z., Goliński, M., Özmen, A.: Modelling Problems in a Regional Labor Market in Poland with MARS (2019)
106. Çevik, A.: Computer-aided diagnosis of Alzheimer's disease and mild cognitive impairment with MARS/CMARS classification using structural MR images (2017)
107. Çevik, A., Weber, G.-W., Eyüboğlu, B.M., Oğuz, K.K., Initiative, A.D.N.: Voxel-MARS: a method for early detection of Alzheimer's disease by classification of structural brain MRI. Ann. Oper. Res. **258**, 31–57 (2017)
108. Kuter, S., Akyurek, Z., Weber, G.-W.: Retrieval of fractional snow covered area from MODIS data by multivariate adaptive regression splines. Remote Sens. Environ. **205**, 236–252 (2018)
109. Kuter, S., Weber, G.-W., Akyürek, Z., Özmen, A.: Inversion of top of atmospheric reflectance values by conic multivariate adaptive regression splines. Inverse Probl. Sci. Eng. **23**(4), 651–669 (2015)
110. Baltas, I., Frangos, N., Yannacopoulos, A.: Optimal investment and reinsurance policies in insurance markets under the effect of inside information. Appl. Stoch. Model. Bus. Ind. **28**(6), 506–528 (2012)
111. Baltas, I., Xepapadeas, A., Yannacopoulos, A.N.: Robust control of parabolic stochastic partial differential equations under model uncertainty. Eur. J. Control. **46**, 1–13 (2019)
112. Savku, E., Weber, G.-W.: Stochastic differential games for optimal investment problems in a Markov regime-switching jump-diffusion market. Ann. Oper. Res. **312**(2), 1171–1196 (2022)
113. Savku, E.: Memory and anticipation: two main theorems for Markov regime-switching stochastic processes. arXiv preprint arXiv:2302.13890 (2023)
114. Ghosh, S., Roy, S.K., Weber, G.-W.: Interactive strategy of carbon cap-and-trade policy on sustainable multi-objective solid transportation problem with twofold uncertain waste management. Ann. Oper. Res. **326**(1), 157–197 (2023)
115. Pervin, M., Roy, S.K., Sannyashi, P., Weber, G.-W.: Sustainable inventory model with environmental impact for non-instantaneous deteriorating items with composite demand. RAIRO Oper. Res. **57**(1), 237–261 (2023)
116. Belen, S.-C., Weber, G.-W., Ozel, M.E.: A search about why the earth may have already been discovered by the extraterrestrial (s) and their possible stochastic travels over interstellar distances. In: 43rd COSPAR Scientific Assembly. Held 28 January–4 February, vol. 43, p. 1936 (2021)
117. Gürbüz, B., Mawengkang, H., Husein, I., Weber, G.-W.: Rumour propagation: an operational research approach by computational and information theory. Cent. Eur. J. Oper. Res. **30**, 345–365 (2022)
118. Gürbüz, B., Gökçe, A.: An algorithm and stability approach for the acute inflammatory response dynamic model. In: Operations Research: New Paradigms and Emerging Applications, pp. 192–217 (2022)
119. Kriner, M.: Survival analysis with multivariate adaptive regression splines. PhD thesis, LMU (2007)

120. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evolut. Comput. **1**(1), 67–82 (1997)
121. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017)
122. Li, Y., Lin, X., Liu, J.: An improved gray wolf optimization algorithm to solve engineering problems. Sustainability **13**(6), 3208 (2021)
123. Wilcoxon, F.: Individual comparisons by ranking methods. In: Breakthroughs in Statistics: Methodology and Distribution, New York, NY: Springer New York, pp. 196–202 (1992)
124. Mohapatra, P., Roy, S., Das, K.N., Dutta, S., Raju, M.S.S.: A review of evolutionary algorithms in solving large scale benchmark optimisation problems. Int. J. Math. Oper. Res. **21**(1), 104–126 (2022)
125. Arora, J.S.: Introduction to Optimum Design. Elsevier. (2004)
126. Sandgren, E.: Nonlinear integer and discrete programming in mechanical design optimization (1990)
127. Sandgren, E.: Nonlinear integer and discrete programming in mechanical design. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 26584, pp. 95–105. American Society of Mechanical Engineers (1988)
128. Nowacki, H.: Optimization in pre-contract ship design (1973)