Journal of Bionic Engineering (2024) 21:1465–1495
https://doi.org/10.1007/s42235-024-00493-8

RESEARCH ARTICLE

Multi-trial Vector-based Whale Optimization Algorithm

Mohammad H. Nadimi-Shahraki[1,2] · **Hajar Farhanginasab**[1,2] · **Shokooh Taghian**[1,2] · **Ali Safaa Sadiq**[3] · **Seyedali Mirjalili**[4]

Received: 5 August 2023 / Revised: 12 February 2024 / Accepted: 19 February 2024 / Published online: 26 April 2024

Abstract
The Whale Optimization Algorithm (WOA) is a swarm intelligence metaheuristic inspired by the bubble-net hunting tactic of humpback whales. In spite of its popularity due to simplicity, ease of implementation, and a limited number of parameters, WOA's search strategy can adversely affect the convergence and equilibrium between exploration and exploitation in complex problems. To address this limitation, we propose a new algorithm called Multi-trial Vector-based Whale Optimization Algorithm (MTV-WOA) that incorporates a Balancing Strategy-based Trial-vector Producer (BS_TVP), a Local Strategy-based Trial-vector Producer (LS_TVP), and a Global Strategy-based Trial-vector Producer (GS_TVP) to address real-world optimization problems of varied degrees of difficulty. MTV-WOA has the potential to enhance exploitation and exploration, reduce the probability of being stranded in local optima, and preserve the equilibrium between exploration and exploitation. For the purpose of evaluating the proposed algorithm's performance, it is compared to eight metaheuristic algorithms utilizing CEC 2018 test functions. Moreover, MTV-WOA is compared with well-stablished, recent, and WOA variant algorithms. The experimental results demonstrate that MTV-WOA surpasses comparative algorithms in terms of the accuracy of the solutions and convergence rate. Additionally, we conducted the Friedman test to assess the gained results statistically and observed that MTV-WOA significantly outperforms comparative algorithms. Finally, we solved five engineering design problems to demonstrate the practicality of MTV-WOA. The results indicate that the proposed MTV-WOA can efficiently address the complexities of engineering challenges and provide superior solutions that are superior to those of other algorithms.

Keywords Swarm intelligence algorithms · Metaheuristic algorithms · Optimization · Engineering design problems · Whale optimization algorithm

1 Introduction

The development of science and technology has led to an increase in the complexity of optimization problems, and the emergence of new optimization problems has necessitated the deployment of the most appropriate optimization algorithms. Deterministic algorithms are successful when dealing with linear, convex, and simple optimization problems; nevertheless, these methods are inefficient when handling non-differentiable objective functions, nonlinear search spaces, non-convex, complicated, and NP-hard issues [1, 2]. On the other hand, these are the key features that optimization issues exhibit in real applications. As a result of the inefficiency of deterministic algorithms, stochastic algorithms, including metaheuristic algorithms, were developed [3]. Metaheuristic algorithms that employ random operators, trial-and-error methods, and random exploration of the search space are effective tools for tackling optimization issues. The widespread usage of metaheuristic algorithms can be attributed to their basic concepts and straightforward implementations, as well as their effectiveness in solving high-dimensional problems [4, 5].

✉ Mohammad H. Nadimi-Shahraki
nadimi@iaun.ac.ir

1 Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad 8514143131, Iran

2 Big Data Research Center, Najafabad Branch, Islamic Azad University, Najafabad 8514143131, Iran

3 Department of Computer Science, Nottingham Trent University, Nottingham NG11 8NS, UK

4 Centre for Artificial Intelligence Research and Optimisation, Torrens University, Brisbane 4006, Australia

Metaheuristic Algorithms (MAs) have been proposed to handle non-linear, multimodal, and high-dimensional optimization problems [6–8]. Using MAs to tackle complicated problems has shown to be an effective alternative to conventional optimization algorithms [9, 10]. Although approximation algorithms such as MAs are not guaranteed to produce the best solution, they are developed to provide solutions as close as possible to the optimal one in a reasonable period of time [1, 11]. The early phases of the search are devoted to exploring the search space, and the promising regions are then exploited in later iterations to improve the quality of the solutions. In addition, by utilizing multiple search agents, these algorithms demonstrate superior performance in avoiding local minima and finding near-optimum solutions [12, 13].

Among the various categories of MAs, evolutionary and swarm intelligence algorithms stand out as the most prominent and have been effectively applied to various real-world challenges. Evolutionary algorithms simulate natural evolution by adapting reproduction, crossover, and mutation operators, whereas swarm intelligence algorithms imitate the collective intelligence of natural groupings, such as birds' flocks, fish's schools, and ants' colonies. Some of the well-known evolutionary algorithms are Genetic Algorithm (GA) [14], Differential Evolution (DE) [15], and Evolution Strategies (ES) [16]. The act of seeking food and the strategies of fighting and hunting that occur naturally among creatures provided a fundamental motivation for the development of a variety of swarm intelligence algorithms, such as Particle Swarm Optimization (PSO) [17], Bat Algorithm (BA) [18], Cuckoo Search (CS) [19], Krill Herd (KH) [20], Grey Wolf Optimizer (GWO) [21], Moth-Flame Optimization (MFO) [22], Butterfly Optimization Algorithm (BOA) [23], Salp Swarm Algorithm (SSA) [24], Honey Badger Algorithm (HBA) [25], and Liver Cancer Algorithm (LCA) [26].

The Whale Optimization Algorithm (WOA) [27] is a swarm intelligence algorithm that emulates humpback whales' intelligence bubble-net hunting behavior. The WOA's simplicity, ease of implementation, and few parameters have attracted many researchers to use it for solving optimization problems, including intrusion detection systems [28], disease detection [29], robotics [30], and signal processing [31]. However, WOA tends to be trapped in local minima due to a deficiency in maintaining a balanced exploration and exploitation. This is because, in the early iterations, WOA merely conducts global exploration and entirely switches to local exploitation, reducing the balance between exploration and exploitation. Due to the absence of global exploration in later iterations, the population leads to fast convergence toward local optima without ensuring global optimality with poor solution accuracy [32, 33]. Therefore, a number of adjustments have been made to the canonical WOA in order to address these flaws [34, 35]. According to the No-Free Lunch (NFL) theorem [36], there is no algorithm that is superior to all other algorithms for dealing with problems with various challenges. Thus, it is required to suggest new algorithms or make improvements to those already in use by altering their operators to tackle optimization issues more efficiently.

The WOA concept, while straightforward, holds the potential to emerge as a leading optimization algorithm. This has led researchers to give it increased attention, leading to numerous enhancements and diverse applications of WOA. Despite the significant improvements of WOA, it still needs more developments that can handle problems with complex characteristics. Therefore, the proposed modifications aim to enhance the overall performance of the WOA, specifically addressing issues where the algorithm demonstrates suboptimal behavior or slow convergence rates. These changes introduce adaptability features, enabling the algorithm to navigate various problem landscapes more effectively. This adaptability becomes crucial, especially when the traditional WOA encounters challenges in efficiently discovering optimal solutions. Moreover, the introduced modifications enhance the WOA's versatility, extending its applicability across various optimization problems.

In our earlier research [37], we developed the Multi-trial Vector (MTV) approach to address an extensive range of optimization issues. The MTV approach includes the following four components: winner-based distributing, multi-trial vector producing, evaluating and population updating, and lifetime archiving. This approach incorporates several search strategies by specifying distribution policies across the population to improve the algorithms' performance. To prevent getting stuck in a local optimum, to prevent the search from converging too quickly, and to strike a proper balance between exploiting and exploring solutions, the MTV approach allows for the definition of multiple strategies that can be adapted to the particular problem characteristics at each stage. The study aims to improve the efficiency of the WOA in addressing complex real-world optimization problems by introducing three new additional trial vectors that leverage the advantages of the MTV approach.

This paper introduces a Multi-trial Vector-based Whale Optimization Algorithm (MTV-WOA) that utilizes the Multi-trial Vector (MTV) approach. The MTV-WOA employs three new Trial Vector Producers (TVPs) during the multi-trial vector producing step of the MTV approach, each designed to address optimization challenges with varying characteristics while preserving specific search behavior. The three proposed TVPs include a Balancing Strategy-based Trial-vector Producer (BS_TVP), a Local Strategy-based Trial-vector Producer (LS_TVP), and a Global Strategy-based Trial-vector Producer (GS_TVP). The MTV approach's winner-based distributing is utilized to apply each TVP to a subset of the population corresponding to that TVP. In MTV-WOA, incorporating the MTV approach involves adopting a winner-based distributing policy. This policy leads to dividing the main population into three distinct subpopulations. The rationale behind this design is rooted in the advantages of the MTV approach, specifically in enhancing exploration and exploitation capabilities. The winner-based distribution strategy allows for a more effective and dynamic allocation of resources across these subpopulations, contributing to the algorithm's overall performance. The proportion of the dedicated population is adjusted regularly based on the number of whales each TVP improves. Additionally, the MTV-WOA utilizes a lifetime archive to store inferior whales to transfer their knowledge to future generations of whales. The proposed algorithm is designed to enhance the performance of the WOA algorithm when applied to complex real-world optimization problems.

The proposed MTV-WOA's performance is validated using 29 benchmark functions of CEC 2018 [38] in 10, 30, and 50 dimensional space. The gained results are compared to state-of-the-art and newly proposed metaheuristic algorithms consists of Krill Herd (KH) [20], Grey Wolf Optimizer (GWO) [21], Moth-Flame Optimization (MFO) [22], Whale Optimization Algorithm (WOA) [27], Salp Swarm Algorithm (SSA) [24], Harris Hawks Optimization (HHO) [39], Butterfly Optimization Algorithm (BOA) [23], and Arithmetic Optimization Algorithm (AOA) [40]. Afterwards, the gained results on benchmark functions by MTV-WOA and comparative algorithms are statistically analyzed by use of the Friedman test [41] to confirm the superiority of the proposed algorithm. In another experiment set against CEC 2017 winners LSHADE-SPACMA [42], LSHADE-cnEpSin [43], well-established algorithm PSO [17], recent algorithms Snake Optimizer (SO) [44] and Coati Optimization Algorithm (COA) [45], and improved variant enhanced whale optimization algorithm (E-WOA) [46], the proposed MTV-WOA consistently outperforms these algorithms, securing its position as the third-best algorithm after the CEC winners. Results, reinforced by the Wilcoxon signed-rank test, highlight MTV-WOA's statistically significant superiority over other algorithms. Furthermore, the study extends its impact by evaluating enhancements introduced by BS_TVP, LS_TVP, and GS_TVP on other algorithms, such as PSO and LSHADE-SPACMA, as demonstrated by the effectiveness of Adapted-PSO and Adapted-LSHADE-SPACMA. This analysis provides insights into the broader applicability of the proposed TVPs in enhancing the performance of diverse optimization algorithms.

Furthermore, the application of MTV-WOA was proved through the resolution of engineering issues. The proposed MTV-WOA demonstrates superiority over comparative algorithms, as evidenced by thorough comparisons and statistical analyses. Equipping WOA with multi-movement strategies significantly strengthens its effectiveness in solving diverse and complex optimization problems, particularly shifted or rotated problems. The advantage of the proposed improved algorithm lies in its ability to enhance the WOA's performance. Leveraging the MTV approach allows simple metaheuristic algorithms like WOA to integrate complementary search strategies. This adaptation uniquely positions the MTV-WOA to excel in addressing various optimization challenges.

The outline of the paper is as follows: Sect. 2 provides a literature survey of relevant works, while the WOA's mathematical model and flowchart are presented in Sect. 3. The proposed MTV-WOA is presented in Sect. 4. Section 5 presents the experimental assessment and statistical analysis of the proposed and comparative algorithms, while Sect. 6 showcases the solution to engineering problems. Section 7 delves into the primary factors contributing to the achievements of the MTV-WOA. The final section summarizes the findings and suggestions for further research.

2 Related Work

Metaheuristic algorithms are popular and powerful algorithms that have been proposed to provide near-optimal solutions to real-world problems. MAs can be categorized into four main groups evolutionary, swarm intelligence, physics-based, and human behavior-based according to the inspiration source. Evolutionary Algorithms (EAs) draw inspiration from Darwin's theory, which simulates the evolutionary behaviors of living things by utilizing concepts of competence and survival. Evolutionary algorithms rely on mechanisms such as mutation and crossover to ensure the best possible solutions survive and evolve. This category's popular algorithms are Evolution Strategy (ES), Genetic Algorithm (GA), Differential Evolution (DE), and Genetic Programming (GP). Among these, DE and its variants' effectiveness and performance have been demonstrated in a number of studies, particularly in a variety of disciplines, such as medical [47], engineering [48], industry [49], economics [50], and data mining [51].

The behavioral model of animals, plants, and birds serves as the basis for Swarm Intelligence (SI) algorithms classified under the second category of metaheuristics. SI algorithms rely on influential population members to guide other solutions to reach the optimal solution. The most popular and recently proposed SI algorithms are Particle Swarm Optimization (PSO) [17], Krill Herd (KH) [20], Grey Wolf Optimizer (GWO) [21], Whale Optimization Algorithm (WOA) [27], Butterfly Optimization Algorithm (BOA) [23], Salp Swarm Algorithm (SSA) [24], Monarch Butterfly Optimization (MBO) [52], Chameleon Swarm Algorithm (CSA) [53], Horse Herd Optimization Algorithm (HOA) [54], Orca Predation Algorithm (OPA) [55], White Shark Optimizer (WSO) [56], Snake Optimizer (SO) [44], and Artificial Hummingbird Algorithm (AHA) [57]. Even though the vast majority of

SI algorithms are intended to deal with continuous problems, a variety of techniques can be used to adapt these algorithms to deal with difficulties of a discrete nature [58]. Utilizing the adapting techniques has allowed for the successful resolution of a number of real-world problems.

In physics-based algorithms, individuals' movements and relationships are modeled by applying physical laws, including gravity, inertia force, and electrical charges. The Big Bang Big-Crunch (BB-BC) [59] is a well-known physics-based algorithm inspired by the big bang and crisis theory. The coulomb law of physics and Newtonian mechanical motion led to the design of Charged System Search (CSS) [60] algorithm. Some other physics-based algorithms are Ray Optimization (RO) [61], Colliding Bodies Optimization (CBO) [62], Atom Search Optimizer [63], Nuclear Reaction Optimization (NRO) [64], and Plasma Generation Optimization (PGO) [65]. Human cultural and political activities such as learning, competitiveness, political campaigns, and cultural influence inspire algorithms based on human behavior. Teacher Learning Based Optimization (TLBO) [66] is a practical example of these algorithms which is model teaching and learning behavior between humans. Poor and Rich Optimization (PRO) [67], Seeker Optimization Algorithm (SOA) [68], Dual-Population Social Group Optimization (DPSGO) [69], and Human Eye Vision Algorithm (HEVA) [70] are well-known and recently proposed physics-based algorithms.

Numerous real-world problems in continuous and discrete domains have been resolved using metaheuristic algorithms, such as image segmentation [71–74], feature selection [75–80], solar power system optimization [81–84], engineering [85–88], planning and scheduling [89–92], disease diagnosis [93–95], continuous optimization problems [96–104], optimal power flow [105, 106], routing problem [107, 108], community detection [109–111], and cloud manufacturing [112–114]. Among the population-based metaheuristic algorithms, the WOA is well-known and has been used in various applications. Meanwhile, WOA possesses significant flaws, including inadequate exploration and low variability, resulting in local optimum trapping, inability to jump out of local optimal, and poor global searchability. Thus, a variety of variants were suggested to tackle its deficiencies. In the following, some improved variants of the whale optimization algorithm are discussed.

In Ref. [32], WOAGWO, a hybridized GWO with WOA was suggested to address global numerical optimization issues. By improving the exploitation of WOA and preventing stagnation within local optima, the WOAGWO significantly enhanced the performance of WOA. A modified WOA named m-SDWOA was proposed in Ref. [33], which combines modified Symbiotic Organisms Search (SOS) [115] with a mutation strategy from the DE algorithm. The proposed m-SDWOA mitigates the shortcomings of the WOA involving insufficient exploitation and the inability to maintain a steadiness between exploring and exploiting. In Ref. [116], an enhanced WOA integrated with SSA named ESSAWOA was proposed to solve global optimization problems. In ESSAWOA, the lens opposition-based learning strategy was utilized to change the position of search agents. Also, the SSA's convergence parameter and the leader mechanism are used to strengthen the exploitation and maintain diversity. Experiments show that ESSAWOA is more accurate in finding the optimal solution than WOA and SSA.

In Ref. [117], a Laplacian whale optimization algorithm was developed known as LXWOA. A Laplace crossover operator was utilized to enhance the WOA algorithm's population variability and address the issue of early convergence that arises during the optimization phase. The results of the experiments demonstrated that the suggested algorithm converges faster than comparative algorithms. In Ref. [118], an enhanced whale optimization algorithm was proposed to overcome the WOA's flaws, such as fast convergence to a local optimum, low computation accuracy, and stagnation. Levy flight strategy and ranking-based mutation operator were added to the WOA to improve the global and local search abilities. Experimental results show that the proposed algorithm has a fast convergence speed and high calculation accuracy.

In Ref. [119], IWOSSA which is a hybridized improved WOA with SSA, was proposed to solve optimization problems. In this regard, IWOA presented a variation of WOA that uses exponential relations rather than linear ones. Then, IWOA or SSA conducts the search based on a particular condition. Experiments using benchmark functions and PID controllers demonstrated that the suggested IWOSSA could achieve superior results and fine-tune the engineering problem's parameters. To address the inadequacies of WOA in tackling high-dimensional problems, a hybrid WOA with several techniques was developed [34]. The proposed algorithm uses individual learning instead of learning dimensions, as well as a random opposition learning strategy to enable the algorithm to find the desired solution in high dimensions. The gained results from diverse experiments have demonstrated that the suggested algorithm effectively solves benchmark functions and clusters high-dimensional datasets.

HS-WOA and HS-WOA + are two algorithms that were proposed in Ref. [120] such that they are the hybridization of WOA and a human-based algorithm Social Group Optimization (SGO) [121]. The suggested algorithm finds the optimal balance between exploration and exploitation by integrating the capabilities of WOA and SGO, which are primarily focused on convergence and exploitation, respectively. Experimental results prove that the hybrid performance is more efficient than the WOA. In Ref. [122], a new variant of WOA named OBCWOA was proposed that uses a chaos mechanism based on quasi-opposition. The purpose of the improved version is to overcome the poor convergence speed of the original WOA and to prevent becoming stuck in a local optimum while dealing with problems with a high dimension. In order to speed up the convergence and create initial values, OBCWOA takes advantage of the turbulence mechanism. In opposition-based learning, balancing exploration with the development of an algorithm to get out of local optimizations is achieved by applying the opposition-based learning approach.

In Ref. [123], an improved WOA with a joint search mechanism named JSWOA was proposed to tackle the high-dimensional optimization problems. In the proposed algorithm, the initial population diversity is maintained using a tent chaotic map. Then, an inertia weight is utilized to boost the convergence speed and escape from the local optima of the JSWOA. A final opposition-based learning mechanism is utilized to constantly upgrade the population's members throughout each iteration in order to improve the quality and variety of the whale population and raise the chance of reaching a globally optimum solution. The proposed JSWOA was evaluated by benchmark functions and the gained results prove the better performance in terms of solution accuracy and convergence speed. Another modified whale optimization algorithm named MWOA-CEE was proposed in Ref. [124] such that the proposed algorithm is suitable for tackling WOA's flaws. In this regard, the proposed MWOA-CEE algorithm utilized three operators consisting of opposition-based learning, exponentially decreasing function, and elite-guided Cauchy mutation. The suggested MWOA-CEE was assessed using benchmark functions, and the results demonstrate its higher solution precision.

A modified whale optimization algorithm with a cross-optimization algorithm named MWOA-CS [125] was suggested for large-scale optimization issues. Random execution of the WOA or cross-optimization algorithm is used to update each problem dimension during the search process. The exploitation and exploration capabilities are enhanced by using the improved WOA algorithm's new nonlinear convergence coefficient and nonlinear weight of inertia. The findings, obtained by evaluating the proposed and comparative algorithms on test functions with dimensions ranging from 300 to 1000, demonstrated that the MWOA-CS provided superior performance compared to other algorithms. In Ref. [126], a multi-strategy whale optimization algorithm named MSWOA was proposed for solving complex engineering optimization problems. In this regard, a high-quality initial population is formed by employing a random chaotic logistics map, and the balance between exploitability and exploration is maintained using adaptive weight modification. Additionally, using a Lévy flight ensures that the population diversity is preserved during each iteration. The effectiveness of the algorithm was demonstrated through experiments conducted on the CEC 2017 benchmark set and by comparisons with other algorithms.

3 Whale Optimization Algorithm (WOA)

The WOA [27] is one of the population-based algorithms belongs to the category of swarm intelligence algorithms which is inspired by humpback whales' natural hunting behavior. In nature, a humpback whale stalks krill or tiny fish near the water's surface using bubble-net hunting strategy. Supposing there is a population of N whales as $\boldsymbol{X} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_N\}$ in a search space and each whale characterized with a D-dimensional vector $\boldsymbol{X}_i = \{x_1, x_2, \ldots, x_D\}$. In WOA, whales estimate the position of their prey which is the position of the best candidate solution or the closest candidate to the optimal solution. Consequently, population members modify their position based on the location of the prey. The WOA simulates three types of hunting behavior in whales, including encircling prey, bubble-net attacking, and seeking prey, as described below and its flowchart shown in Fig. 1.

Encircling Prey: The initial stage of a whale's hunting strategy is to encircle its prey. During this phase, the whales have located their prey and are closing in for the attack. In the algorithm, $\boldsymbol{X}^*$, the current best whale, is regarded as the prey, and other whales modify their positions relative to this position using Eqs. (1) and (2).

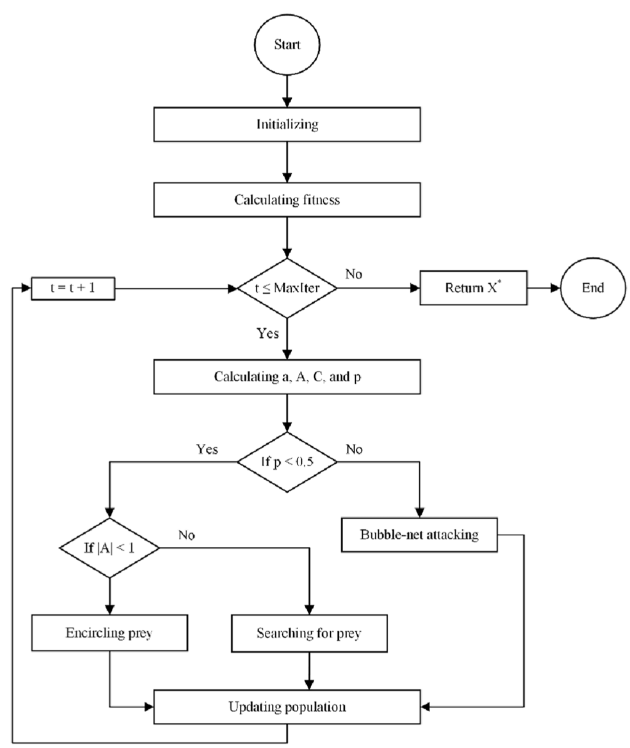


Fig. 1 The flowchart of WOA

$$D = |C \times X^*(t) - X(t)| \tag{1}$$

$$X(t+1) = X^*(t) - A \times D \tag{2}$$

where $\boldsymbol{D}$ is the estimated distance between $\boldsymbol{X}^*$ and $\boldsymbol{X}$ in the t-th iteration, C and A are coefficients determined by Eqs. (3) and (4), respectively.

$$A = 2 \times a \times r - a \tag{3}$$

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

where a is gradually decreased from two to zero through the iterations, and r is a random number in the range zero and one.

Bubble-Net Attacking: As the whales whirl around their prey, they update their position in a spiral pattern or engage in a shrinking encirclement strategy. The spiral position updating is modeled in Eq. (5),

$$X(t+1) = D' + e^{bl} \times \cos(2\pi l) + X^*(t) \tag{5}$$

where $\boldsymbol{D'}$ is the distance between the position of whale $\boldsymbol{X}$ and the position of $\boldsymbol{X}^*$ calculated by Eq. (6), b is a constant coefficient by value set to one, and l is a uniform random number in range $[-1, +1]$.

$$D\prime = |X^*(t) - X(t)| \tag{6}$$

The bubble-net attacking is modeled by Eq. (7),

$$X(t+1) = \begin{cases} X^*(t) - A \times D & if\ p < 0.5 \\ D\prime \times e^{bl} \times \cos(2\pi l) + X^*(t) & if\ p \geq 0.5 \end{cases} \tag{7}$$

where the variable p is a random integer that ranges from 0 to 1 and is used to determine the likelihood of updating whale positions using either the shrinking encircling strategy (when p is less than 0.5) or the spiral updating technique (when p is greater than or equal to 0.5). The shrinking encircling strategy involves a random variable A, which ranges from -a to a, where the value of a linearly decreases from two to zero with each iteration. The spiral updating strategy, on the other hand, relies on a distance measuring $\boldsymbol{D'}$ that indicates the distance between $\boldsymbol{X}$ and $\boldsymbol{X}^*$ in the spiral updating position. The constant b determines the form of the spiral movement, while the variable l is a random value between -1 and 1.

Searching for Prey: Whales search the whole search space to find potential prey. When $|A| \geq 1$, a whale conduct global search or exploration by using the search for prey strategy. During the exploration phase, which is determined by Eqs. (8) and (9), the whale shifts its location relative to a random whale $\boldsymbol{X}_{\boldsymbol{rand}}$ instead of the best whale $\boldsymbol{X}^*$:

$$D = |C \times X_{rand} - X(t)| \tag{8}$$

$$X(t) = |X_{rand} - A \times D| \tag{9}$$

where $\boldsymbol{X}_{\boldsymbol{rand}}$ is a randomly chosen whale from the current population.

4 Multi-Trial Vector-Based Whale Optimization Algorithm (MTV-WOA)

The WOA is a widely used optimization algorithm with a straightforward implementation; however, the algorithm's performance is insufficient when dealing with complex problems. The canonical WOA suffers from low exploration and slow convergence speed, which significantly affect its performance [122]. The canonical WOA's performance can be improved by altering its search strategy that incorporates multiple search strategies when dealing with complex problems with diverse characteristics. Motivated by this, a Multi-trial Vector-based Whale Optimization Algorithm (MTV-WOA) is proposed such that the simple WOA search strategy is replaced by the Multi-trial Vector (MTV) approach [37]. Integrating the MTV approach into the WOA facilitates the development of a variety of Trial Vector Producers (TVPs) so that each can maintain a distinct behavior throughout the optimization process. Additionally, according to MTV's winner-based distributing policy, each TVP is applied to a specific portion of the population. Thus, information sharing between whales from distinct subpopulations during population dispersion can increase the effectiveness of the proposed algorithm.

As depicted in Fig. 2, the MTV-WOA has five steps consists of: initializing, winner-based distributing, multi-trial vector producing, evaluating and population updating, and lifetime archiving. Following initializing N whales in the search space, the subpopulation size of each TVP is calculated and altered within every sections of iterations named *WinIter* in the winner-based distributing step. After initializing the main population (X), the population is then partitioned into three subpopulations, namely *X_BT*, *X_LT*, and *X_GT*. Each of these subpopulations corresponds to one of the TVPs. This partitioning ensures a diverse and balanced distribution of whales across the subpopulations, facilitating a more comprehensive exploration of the search space. Then, in the multi-trial vector producing step, for each whale a candidate position is produced according to one of the three trial vector producers. In MTV-WOA, three new search strategies are proposed, namely Balancing Strategy-based Trial-vector Producer (BS_TVP), Local Strategy-based Trial-vector Producer (LS_TVP), and Global Strategy-based Trial-vector Producer (GS_TVP). These new search strategies are

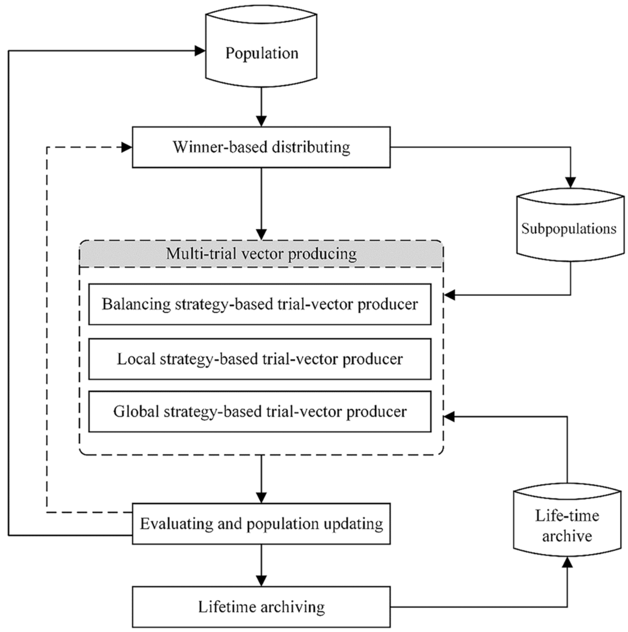


Fig. 2. The model of the proposed Multi-trial Vector-based Whale Optimization Algorithm (MTV-WOA)

incorporated to prevent local optima entrapment, increase exploration and exploitation, and maintain a balance between them. BS_TVP ensures an equilibrium between exploration and exploitation, and avoidance of the optimal local solution, LS_TVP enhances the exploitation ability, and GS_TVP promotes exploration. Finally, in the evaluating and population updating step inferior whales preserves in a lifetime archive to use their information to propagate the current population. Table 1 gives a nomenclature of the used parameters in the proposed algorithm. The following section provides a detailed explanation of the algorithm's steps.

Initializing Step: In the proposed algorithm, the whale's position is represented by a vector $\boldsymbol{X}_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,D}\}$ where $\boldsymbol{X}_i$ is the position of i-th whale and D is the problem's number of dimensions. Between the search space's lower and upper limit borders, N whales are distributed at random by Eq. (10).

$$x_{i,j} = L_j + \left(U_j - L_j\right) \times rand(0, 1) \tag{10}$$

where $x_{i,j}$ is the j-th dimension of the i-th whale, L_j and U_j are the lower and upper bound values of the j-th dimension, and *rand* is the random number generated in the range [0, 1], respectively. N-generated whales' position is stored in matrix $\boldsymbol{X}_{N\times D}$ shown in Eq. (11).

$$X^t = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,D} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,D} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,D} \end{bmatrix} \tag{11}$$

On the t-th iteration, the objective function $f(\boldsymbol{X}^t_i)$ assesses the fitness value of each whale $\boldsymbol{X}^t_i$, respectively.

Winner-Based Distributing Step: Through this step, the number of iterations is split into k WinIter sections, each of which contains *nIt* iterations. The winning TVP is the TVP with the highest value of improved rate in the previous WinIter. The subpopulation size of each TVP is calculated at the conclusion of each WinIter by calculating the improved rate given by *ImpRate* using Eq. (12).

$$ImpRate_{BS\_TVP} = \frac{IF_{BS\_TVP}}{FE_{BS\_TVP}} \tag{12}$$

$$ImpRate_{LS\_TVP} = \frac{IF_{LS\_TVP}}{FE_{LS\_TVP}}$$

$$ImpRate_{GS\_TVP} = \frac{IF_{GS\_TVP}}{FE_{GS\_TVP}}$$

where $ImpRate_{BS\text{-}TVP}$, $ImpRate_{LS\text{-}TVP}$, and $ImpRate_{GS\text{-}TVP}$ are the improved rate calculated for each TVP. IF_{BSTVP}, $IF_{LS\text{-}TVP}$, and $IF_{GS\text{-}TVP}$ are the number of whales whose fitness is improved by BS_TVP, LS_TVP, and GS_TVP. Also, $FE_{BS\text{-}TVP}$, $FE_{LS\text{-}TVP}$, and $FE_{GS\text{-}TVP}$ are the number of function evaluations in the previous WinIter. In the MTV-WOA, the reward-penalty distribution policy is considered to determine the subpopulation size of each TVP by using Eqs. (13) and (14),

Table 1 The nomenclature used in MTV-WOA

Parameter	Description
X	The whales' population
M, $\overline{M}$	The transformation matrix and its reverse
$ImpRate_{BS\text{-}TVP}$, $ImpRate_{LS\text{-}TVP}$, $ImpRate_{GS\text{-}TVP}$	The improved rate of BS_TVP, LS_TVP, and GS_TVP
X_BT, X_LT, X_GT	The subpopulation of BS_TVP, LS_TVP, and GS_TVP
C_BT, C_LT, C_GT	The candidate subpopulation of BS_TVP, LS_TVP, and GS_TVP
X_Upop	The union population of $\boldsymbol{X}$ and lifetime archive

$$Reward\ rule:\ If\ BS\_TVP\ or\ LS\_TVP\ is\ Win-TVP,\ then\ N_{Win-TVP} = 0.6 \times N\ and\ N_{Loser-TVPs} = 0.2 \times N \tag{13}$$

$$Penalty\ rule:\ If\ GS\_TVP\ is\ Win-TVP,\ then\ N_{Win-TVP} = 0.2 \times N\ and\ N_{Loser-TVPs} = 0.4 \times N \tag{14}$$

where $N_{Win\text{-}TVP}$, $N_{Loser\text{-}TVP}$ and N are subpopulation size of winner TVP, loser TVP and the total number of whales, respectively.

Multi-Trial Vector Producing Step: In the proposed algorithm, in each iteration, the position of whale X_i changes by one of the proposed strategies including Balancing Strategy-based Trial-vector Producer (BS_TVP), Local Strategy-based Trial-vector Producer (LS_TVP), and Global Strategy-based Trial-vector Producer (GS_TVP). Specifically, the BS_TVP strikes an equilibrium between exploration and exploitation, the LS_TVP is designed to boost exploitation capability, and the GS_TVP provides a significant capability for enhancing the exploration of the proposed MTV-WOA.

Some preliminary information is presented first, followed by a comprehensive explanation of the proposed TVPs. In the proposed BS_TVP and GS_TVP, two transformation matrices $\boldsymbol{M}$ and $\overline{\boldsymbol{M}}$ are used to generate the trial vectors of each subpopulation. Matrix $\boldsymbol{M}$ with dimensions $N \times D$ is constructed from a $D \times D$ lower triangular matrix with values one, by replicating the square matrix (*N/D*) times. The remaining rows of $\boldsymbol{M}$, if there exist, are filled with the first rows of the square matrix. Then, a random permutation is applied to the rows of $\boldsymbol{M}$. Afterward, by replacing the inverse value of each element of $\boldsymbol{M}$, the $\overline{\boldsymbol{M}}$ matrix is obtained. Moreover, in the proposed LS_TVP and GS_TVP, scale factor $F \in [0, 1]$ is a real number that is utilized for scaling the difference vectors. The Cauchy distribution is employed to generate and update the value of F for each whale [37].

Balancing Strategy-Based Trial-Vector Producer (BS_TVP): Since one of the significant drawbacks of WOA is the inability to strike a proper balance, the BS_TVP is proposed to tackle the imbalance between exploring and exploiting and the need to prevent local optimums.

For each whale $\boldsymbol{X\_BT}_i$ belongs to the subpopulation of BS_TVP, a trial vector $\boldsymbol{V\_BT}_i$ is calculated by Eq. (15),

$$V\_BT_i^{t+1} = p \times \left(X\_BT_{rnd}^t - A \times Dist_i^t\right) + p \times (X\_Upop_{rnd}^t - X\_BT_i^t) \tag{15}$$

where p is a convergence coefficient that decreases from two to zero as the number of iterations increases and is calculated by Eq. (16), $\boldsymbol{Dist}_i$ indicates the distance of the i-th whale and $\boldsymbol{X\_BT}_{rnd}$ a randomly selected whale from $\boldsymbol{X\_BT}$ subpopulation and is calculated by Eq. (17), and coefficient A is calculated by Eq. (3).

$$p = 2 - t \times (2/MaxIter) \tag{16}$$

$$Dist_i^t = X\_BT_{rnd}^t - X\_BT_i^t \tag{17}$$

where t and $MaxIter$ represent the current and maximum number of iterations. $\boldsymbol{X\_BT}_{rnd}$ and $\boldsymbol{X\_BT}_i$ are a randomly selected and the i-th whale from the subpopulation $\boldsymbol{X\_BT}$, respectively. The candidate trial vector of the i-th whale $\boldsymbol{C\_BT}_i$ is calculated by Eq. (18),

$$C\_BT_i^{t+1} = M_i^t \times X\_BT_i^t + \overline{M}_i^t \times V\_BT_i^{t+1} \tag{18}$$

where $\boldsymbol{M}_i$ and $\overline{\boldsymbol{M}}_i$ are corresponding values of the i-th whale and $\boldsymbol{V\_BT}_i^{t+1}$ is the candidate trial vector generated for the i-th whale of BS_TVP subpopulation.

Local Strategy-Based Trial-Vector Producer (LS_TVP): This strategy is designed with the intention of enhancing the exploitation efficiency of the proposed MTV-WOA. The current location of whale $\boldsymbol{X\_LT}_i$ is taken into account when calculating the new trial position for the i-th whale of the LS_TVP subpopulation. Additionally, half of the distance between randomly picked whales from $\boldsymbol{X\_Upop}$ and $\boldsymbol{X\_LT}$ is factored into the calculation. Equation (19) is utilized to produce the candidate trial vector for the LS_TVP whale members.

$$C\_LT_i^{t+1} = X\_LT_i^t + (1 - \omega \times F_i) \times \frac{X\_Upop_{rnd}^t - X\_LT_{rnd}^t}{2} \tag{19}$$

where ω is a constant value set by 2, F_i is the scale factor for the i-th whale of LS_TVP subpopulation, $\boldsymbol{X\_Upop}_{rnd}$ is a random individual selected from the union of current and lifetime archive populations, and $\boldsymbol{X\_LT}_{rnd}$ is a random selected whale from the $\boldsymbol{X\_LT}$ subpopulation.

Global Strategy-Based Trial-Vector Producer (GS_TVP): In introducing this TVP, the objective is to improve the exploration capability of the MTV-WOA. When developing

GS_TVP, we used the advantages provided by the movements of the classical WOA and improved upon those advantages by using the disparity between the best and worst whales in the ***X_GT*** subpopulation. The trial vector of the i-th whale $\boldsymbol{V\_GT_i}$ is calculated by Eq. (20),

$$V\_GT_i^{t+1} = \left(Dist\prime \times e^{bl} \times \cos(2\pi l)\right) + \left(X_{best} - A \times Dist\prime\right) + F_i \times (X\_GT_{best}^t - X\_GT_{worst}^t) \quad (20)$$

where ***Dist'*** is the distance of the i-th whale in the GS_TVP subpopulation from the global best position of the entire population calculated by Eq. (21). The spiral motion parameters b and l are the same as the classical WOA. $\boldsymbol{X_{best}}$, $\boldsymbol{X\_GT_{best}}$ and $\boldsymbol{X\_GT_{worst}}$ are the best member of $\boldsymbol{X}$, and best and worst members of the GS_TVP subpopulation, respectively.

$$Dist\prime = \left|X_{best} - X\_GT_i^t\right| \quad (21)$$

The candidate trial vector of the i-th whale of GS_TVP subpopulation $\boldsymbol{C\_GT_i}$ is calculated by Eq. (22),

$$C\_GT_i^{t+1} = M_i^t \times X\_GT_i^t + \overline{M}_i^t \times V\_GT_i^{t+1} \quad (22)$$

where $\boldsymbol{M_i}$ and $\boldsymbol{\overline{M_i}}$ are corresponding values of the i-th whale, $\boldsymbol{X\_GT_i}$ and $\boldsymbol{V\_GT_i}$ are the i-th member of the ***X_GT*** subpopulation and the generated trial vector for the i-th whale of GS_TVP subpopulation.

Evaluating and Population Updating: During each iteration of the proposed MTV-WOA, the three new trial vectors, BS_TVP, LS_TVP, and GS_TVP, generate candidate trial vectors $\boldsymbol{C\_BT^{t+1}}$, $\boldsymbol{C\_LT^{t+1}}$, and $\boldsymbol{C\_GT^{t+1}}$. The fitness value of each candidate trial vector is then evaluated and compared with the previous fitness value of the whales. If the fitness value of a candidate trial vector is better than the whale's previous fitness value, the whale's position is updated with the candidate trial vector. However, if the fitness value of the candidate trial vector is worse than the whale's previous fitness value, the position of the whale remains unchanged in the population. This process is repeated for each whale in the population at every iteration.

Lifetime Archiving: Since the whales that are updated and replaced by their respective candidate trial vectors offer valuable information about previously explored potential regions, it is advantageous to store them. The lifetime archive is used to preserve inferior whales for the purpose of propagating their information to the next generation of whales in subsequent iterations. This archive is initially empty; however, it is capable of storing the position and lifetime of N inferior whales. The life-time value of archived whales is increased by one at the end of each iteration. When the number of archive members exceeds the maximum size, inferior whales will be discarded, considering the higher value of life-time to limit the number of solutions below the maximum size. MTV-WOA receives a boost to its diversity due to this lifetime archive, which also helps prevent early convergence.

Finally, the search procedure is repeated up until the point where it has reached the maximum number of possible repetitions. The pseudo-code for the proposed MTV-WOA is shown in Algorithm 1.

5 Experimental Evaluation and Results

In this section, the performance of the proposed MTV-WOA is experimentally and statistically evaluated using benchmark functions from CEC 2018 [38] with varying dimensions 10, 30, and 50. The quantitative assessment of the proposed algorithm includes the mean of fitness error. The purpose of these experiments is to demonstrate the exploration and exploitation capabilities and the local optima avoidance of the MTV-WOA. Further, the performance of MTV-WOA was evaluated and compared to that of the state-of-the-art as well as recently developed metaheuristic algorithms including Krill Herd (KH) [20], Grey Wolf Optimizer (GWO) [21], Moth-Flame Optimization (MFO) [22], Whale Optimization Algorithm (WOA) [27], Salp Swarm Algorithm (SSA) [24], Harris Hawks Optimization (HHO) [39], Butterfly Optimization Algorithm (BOA) [23], and Arithmetic Optimization Algorithm (AOA) [40]. Finally, the gained results on benchmark functions by MTV-WOA and comparative algorithms are statistically evaluated by the Friedman test [41] for the purpose of establishing the superiority of the suggested algorithm.

5.1 Benchmark Functions

The proposed algorithm was tested using the benchmark functions of CEC 2018 [38]. Note that the test functions used in CEC 2018 were identical to those in CEC 2017, with the exception that the function F2 was excluded from the comparison. These test functions are challenging and have diverse characteristics. It is worth noting that the CEC 2018 test functions are the same as those of CEC 2017, except for F2, which was excluded. The benchmark functions are categorized into unimodal, multimodal, hybrid, and composition functions. The unimodal functions F1 and F3 are suitable to evaluate the algorithms' exploitation ability since

Algorithm 1 Multi-trial vector-based Whale Optimization Algorithm (MTV-WOA)

Input: N, D, $MaxIter$, nIt
Output: The global optimum ($\boldsymbol{X}_{best}$)
Begin
$t = 1$, *Win-TVP* = BS_TVP.
Randomly distribute N whales in the search space.
Evaluating the fitness $f(\boldsymbol{X}^t_i)$ and set the $\boldsymbol{X}^t_{best}$.
While $t \le MaxIter$
  If mod (t, nIt) == 0
    Determining *Win-TVP* using Eq. (12).
  End if
  Winner-based distributing (*Win-TVP*) using Eqs. (13) and (14).
  Do for each TVP
    Multi-trial vector producing.
  End do
  Evaluating and population updating.
  Lifetime archiving.
  Finding and Updating $\boldsymbol{X}_{best}$.
   $t = t + 1$.
End while

they have one optimum. On the other hand, the multimodal functions F4-F10 contain many local optima, which are used to assess the algorithms' ability to explore the search space and avoid local optima. The hybrid and composition functions are more intricate and difficult than the unimodal and multimodal functions. They assess the algorithm's capacity to solve real-world problems and strike a suitable equilibrium between exploring and exploiting.

5.2 Experimental Setting and Environment

The experiments were conducted on a computer with an Intel Core™ i7-6500U 2.50 GHz processor and 16.00 GB RAM using MATLAB R2018a. The population size and the Maximum Number of Iterations (*MaxIter*) were set to 100 and $D \times 10{,}000/N$, respectively. Each algorithm was run 20 times, and the fitness error (f—f^*) was used to report the results, where f is the fitness value of the optimization achieved by the respective algorithm and f^* is the global best value of the optimization problem. Mean fitness error was used to measure the performance of the algorithms. Table 2 presents the parameter settings of the comparative and suggested MTV-WOA algorithms. The comparative algorithms' parameter values were set according to their respective articles' recommendations. The results of the experiments are tabulated in Tables 8, 9, 10, 11 in Appendix A, with the values of the best-gained error highlighted in bold. The number of wins (w), ties (t), and losses (l) for each algorithm in each dimension are listed in the last three rows of each table, which are labeled "w/t/l".

Table 2 Parameters settings

Algorithms	Setting
KH	$V_f = 0.02$, $D^{max} = 0.005$, $N^{max} = 0.01$
GWO	$a = [2\ 0]$
MFO	$a = [-2\ -1]$
WOA	$a = [2\ 0]$, $l = [-1\ 1]$
SSA	$c_2 =$ rand [0 1], $c_3 =$ rand [0 1]
HHO	$E = [2\ 0]$, r_1, r_2, $r_3 =$ rand [01], $beta = 1.5$
BOA	$p = 0.8$, $a = [0.1\ 0.3]$, $c = 0.01$
AOA	$Alpha = 5$, $Mu = 0.499$
MTV-WOA	$nIt = 20$, $\omega = 2$, $b = 1$, $l = [-1\ 1]$

5.3 Exploration and Exploitation Evaluation

As previously mentioned, unimodal functions are useful for evaluating an algorithm's exploitation capabilities, while multimodal functions are more appropriate for testing exploration capabilities. Thus, these two categories of functions are helpful in assessing an algorithm's exploration and exploitation abilities. The results of the MTV-WOA on the F1 and F3 functions in Table 8 in Appendix A demonstrate that the proposed algorithm produces more competitive results than the canonical WOA and the other comparative algorithms, particularly on the F1 function. Therefore, it concludes that the suggested MTV-WOA can converge to the best possible global solution for problems with a single optimum. Additionally, the results presented in Table 9 confirm that the proposed algorithm outperforms the comparative algorithms in all dimensions when solving the benchmark functions F4-F10, which contain multiple local optima. Due to the GS_TVP's random movements of trapped whales, it can be asserted that the proposed MTV-WOA algorithm has an effective exploration ability based on the provided results and comparisons.

5.4 Local Optima Avoidance Evaluation

In this experimental assessment, a set of benchmark functions are utilized to compare the proposed algorithm with

Table 3 The pressure vessel problem's results

Alg.	Optimum values				Optimum cost
	T_s	T_h	R	L	
KH	0.82249	0.40560	42.50602	171.76174	5978.8767
GWO	0.77835	0.38520	40.32066	200.00000	5888.4298
MFO	0.77817	0.38465	40.31962	200.00000	**5885.3328**
WOA	0.84449	0.51762	43.44377	160.64982	6382.9115
SSA	0.78191	0.38650	40.51336	197.52133	5896.1188
BOA	2.71211	0.67304	66.84383	10.00000	16463.1326
HHO	0.90413	0.44326	46.33028	130.42676	6181.1408
AOA	0.84712	0.48061	40.50221	200.00000	6703.8860
MTV-WOA	0.77817	0.38465	40.31962	200.00000	**5885.3328**

Best results among all algorithms are indicated in bold

Table 4 The three-bar truss problem's results

Alg.	Optimum values		Optimum weight
	x_1	x_2	
KH	0.78836	0.40914	263.8960
GWO	0.78905	0.40719	263.8963
MFO	0.78873	0.40808	**263.8958**
WOA	0.78896	0.40743	263.8959
SSA	0.78550	0.41729	263.8959
BOA	0.78991	0.40708	264.1292
HHO	0.78924	0.40667	263.8961
AOA	0.79423	0.39327	263.9702
MTV-WOA	0.78868	0.40825	**263.8958**

Best results among all algorithms are indicated in bold

Table 5 The welded beam problem's results

Alg.	Optimum values				Optimum cost
	h	l	t	b	
KH	0.20468	3.49450	9.05463	0.20569	1.72924
GWO	0.20553	3.47493	9.03912	0.20572	1.72551
MFO	0.20572	3.47076	9.03662	0.20573	1.72487
WOA	0.19981	3.52844	9.22570	0.20565	1.75554
SSA	0.20633	3.46270	9.02329	0.20634	1.72705
BOA	0.23307	4.85155	6.42587	0.41698	2.72129
HHO	0.20253	3.55228	9.00765	0.20777	1.74135
AOA	0.19653	3.37508	10.00000	0.20435	1.85219
MTV-WOA	0.20573	3.47049	9.03662	0.20573	**1.72485**

Best results among all algorithms are indicated in bold

other algorithms in terms of their ability to avoid local optima and balance exploration and exploitation, taking into consideration the results obtained from hybrid and composition functions. Table 10 demonstrates that MTV-WOA outperforms comparative algorithms for solving hybrid functions in three dimensions. The primary reason for this superiority is the exploitation efficiency provided by LS_TVP, which, in combination with BS_TVP, ensures an optimal balance between local and global search, preventing premature convergence of the functions. Additionally, Table 11 presents detailed results of composition functions, where MTV-WOA achieves superior results compared to other algorithms. MTV-WOA achieves an equilibrium between discovery and extraction by utilizing each TVP's improved rate to determine subpopulation sizes. The results indicate that MTV-WOA strikes an appropriate balance between exploring and exploiting, enhancing its ability to avoid local optima in composition functions. It is inferred that the proposed MTV-WOA effectively balances exploration and exploitation. Based on the gained results presented in Tables 10 and 11, it can be deduced that MTV-WOA in dimensions 10, 30, and 50 is more effective than the comparative algorithms.

5.5 Convergence Evaluation

In this experiment, the performance of MTV-WOA is analyzed and compared to that of comparative algorithms on various functions of dimensions 10, 30, and 50, to examine their convergence behavior. The convergence curves for MTV-WOA and comparative algorithms are generated based on the average of the best obtained fitness values over 20 runs. The obtained curves for unimodal, multimodal, hybrid, and composition functions are presented in Figs. 3 and 4 in Appendix A.

The analysis reveals that MTV-WOA exhibits diverse convergence characteristics for test functions with different features. Three distinct convergence patterns are observed during the optimization process. Firstly, there is accelerated convergence in the early iterations, followed by abrupt changes during the first half of the generations, indicating the efficient balance between exploration and exploitation. Subsequently, the estimation of the optimal global solution becomes more precise. Finally, a gradual improvement in convergence towards optimal solutions is observed. These convergence patterns demonstrate that MTV-WOA is more effective than the comparative algorithms in establishing an equilibrium between exploring and exploiting through the iterations.

5.6 Statistical Analysis

The experimental evaluation of the proposed MTV-WOA algorithm showed better performance compared to the comparative algorithms. However, the statistical significance of these results has not been established. To demonstrate the statistical superiority of MTV-WOA, the Friedman test [41] was conducted using Eq. (23) to rank the algorithms based on their fitness values.

$$F_f = \frac{12 \times m}{q \times (q+1)} \left[\sum_k p_k^2 - \frac{q \times (q+1)^2}{4} \right] \tag{23}$$

Where q indicates the number of algorithms, m is the number of case tests, and P_k is the mean rank of the k-th algorithm. The ranking was done by calculating the average rank for each algorithm/problem pair, and then determining the final ranking for each algorithm. The algorithm with the smallest overall rank is considered to be better. In Table 12, the results of the Friedman rank test are reported, revealing that the non-parametric test yielded a significant p-value at a 95% confidence level. According to the overall rankings of the algorithms, it was determined that the MTV-WOA outperformed the comparative algorithms in dimensions 10, 30, and 50. Therefore, it can be concluded that the proposed MTV-WOA is statistically significant and superior to the comparative algorithms.

5.7 Impact Analysis of Using the Proposed TVPs

In this section, the analysis focuses on evaluating the individual performance of the proposed search strategies, BS_TVP, LS_TVP, and GS_TVP, and their collective impact on the performance of the MTV-WOA. The results of this experiment are illustrated in Fig. 5 in Appendix B, depicting the algorithm's performance on selected functions across diverse categories within CEC 2018. BS_TVP exhibits steady convergence and low objective values, showcasing its prowess in balancing exploration and exploitation. It effectively prevents entrapment in local optima, reaching optima comparable to MTV-WOA for certain functions. LS_TVP, with its slower but consistent convergence, prioritizes exploration over exploitation, exploring more of the search space and favoring diversity over fine-tuning solutions. GS_TVP undergoes initial fluctuations followed by smoothing, indicating a transition from exploration to exploitation and highlighting enhanced exploitation efficiency with competitive optima reached. MTV-WOA, as a combination of these strategies, adeptly balances both exploration and exploitation. It matches or surpasses individual variants in optimum reached, validating its superior exploration and exploitation capabilities. The faster and more stable convergence of MTV-WOA compared to BS_TVP, LS_TVP, and GS_TVP algorithms underscores the effective complementary effects of the balance in exploration and exploitation.

5.8 Comparison of MTV-WOA with Well-stablished, Recent, and WOA Variant Algorithms

In this experiment, the proposed MTV-WOA is compared to that of the CEC 2017 winners, LSHADE-SPACMA [42] and LSHADE – cnEpSin [43], well-established algorithm PSO [17], recent algorithms Snake Optimizer (SO) [44] and Coati Optimization Algorithm (COA) [45], and improved variant enhanced whale optimization algorithm (E-WOA) [46]. The experiments conducted here are based on a maximum population size of 428 and minimum size 4 for the LSHADE-SPACMA and LSHADE – cnEpSin. The maximum number of iterations and population size for the other algorithms are set according to their previously defined values. The results of the experiment, presented in terms of mean fitness error, are tabulated in Table 13 in Appendix B. These algorithms were independently applied 20 times to the CEC 2018 test functions with a dimensionality of 10. Moreover, the Wilcoxon signed-rank test is utilized to illustrate the distinction in performance achieved by the proposed MTV-WOA compared to other algorithms [41]. Table 14 presents the outcomes of this pairwise statistical test with a significance level $\alpha = 0.05$. The p-value analysis results confirm that the proposed MTV-WOA's superiority is statistically significant compared to the comparative algorithms. In the convergence analysis of the MTV-WOA algorithm, plotted in Fig. 6, it becomes evident that its convergence curves share striking similarities with those of the LSHADE-SPACMA and LSHADE – cnEpSin algorithms which illustrates their proficiency in adapting their search throughout the iterative optimization process.

5.9 Applicability of the Proposed TVPs for Improving Other Algorithms

In this section, a dedicated experiment set is conducted to show the potential performance improvements of well-established algorithms, such as PSO and DE-based ones, through adapting proposed enhancements. This experiment set investigates the impact of incorporating the proposed BS_TVP, LS_TVP, and GS_TVP with well-established algorithms through the MTV approach. Two new algorithms, Adapted-PSO and Adapted-LSHADE-SPACMA, were meticulously designed for incorporating PSO and LSHADE-SPACMA algorithms with the proposed TVPs. The comparative results, detailed in Table 15 in Appendix B, provide valuable insights into the performance of PSO and DE when enhanced with BS_TVP, LS_TVP, and GS_TVP in the context of MTV. By incorporating these improvements, Adapted-PSO and Adapted-LSHADE-SPACMA are illustrative examples of the adaptability and effectiveness of the proposed enhancements across different optimization algorithms.

6 Solving Engineering Design Problems

In this section, five engineering problems defined in Appendix C were used to test the MTV-WOA's capability for handling actual engineering issues. Pressure vessel [127], three-bar truss [128], welded beam [129], tension/compression spring [130], and speed reducer [131] have all been solved using MTV-WOA and other comparative algorithms. As MTV-WOA is intended to be used for optimization purposes, it should be able to handle the equality and inequality constraints included in these engineering design problems. In this paper, the death penalty function [1] used to handle constraints which is one of the simplest multi-constraint problem-solving procedures among the many constraint-handling methodologies. In order to eliminate infeasible solutions, the death penalty function provides a high fitness value to solutions that break one or more restrictions. In this experiment, each algorithm is executed 30 times, with N and $MaxIter$ set to 20 and $(D \times 10^4)$ / N, respectively. The results of the engineering design problems shown in Table 3, 4, 5, 6, 7 indicate that the MTV-WOA is better to other methods for addressing real-world mechanical engineering issues.

Table 6 The tension/compression spring design problem's results

Alg.	Optimum values			Optimum weight
	d	D	N	
KH	0.051766	0.358574	11.181423	0.012666
GWO	0.050926	0.338577	12.442653	0.012682
MFO	0.051705	0.357113	11.265845	**0.012665**
WOA	0.052225	0.369739	10.564506	0.012670
SSA	0.050000	0.315082	14.343035	0.012668
BOA	0.050000	0.311363	15.000000	0.013233
HHO	0.052122	0.367235	10.698016	0.012669
AOA	0.050000	0.310446	15.000000	0.013194
MTV-WOA	0.051694	0.356828	11.282512	**0.012665**

Best results among all algorithms are indicated in bold

7 Discussion

This study introduced the MTV-WOA algorithm across diverse dimensions and function categories, showcasing its superior performance in unimodal and multimodal functions, as indicated by the results in Tables 8 and 9. MTV-WOA excels in hybrid and composition functions based on the reported results in Tables 10 and 11, leveraging specific search strategies for efficient exploration and exploitation. The Friedman test's overall rankings across dimensions 10, 30, and 50 establish the clear superiority of MTV-WOA, supported by its statistical significance. In the comparative analysis, MTV-WOA outperforms well-established, recent, and WOA variant algorithms, positioning it as the third-best algorithm after the CEC winners, LSHADE-SPACMA and LSHADE – cnEpSin algorithms, in the Friedman test. The Wilcoxon signed-rank test further proves MTV-WOA's superiority, revealing statistically significant differences from other algorithms. The convergence analysis in Fig. 6 in Appendix B illustrates remarkable similarities with LSHADE-SPACMA and LSHADE – cnEpSin algorithms, emphasizing their proficiency in balancing exploration and exploitation. Beyond MTV-WOA, the study assesses the

Table 7 The speed reducer problem's results

Alg.	Optimum values							Optimum cost
	b	m	p	l_1	l_2	d_1	d_2	
KH	3.5001971	0.700013	17.000	7.301146	7.716256	3.350220	5.286671	2994.65080
GWO	3.5002272	0.700000	17.000	7.490734	7.780589	3.350865	5.286804	2997.93584
MFO	3.5	0.700000	17.000	7.300000	7.715320	3.350215	5.286654	**2994.47107**
WOA	3.517224	0.700000	17.000	7.300000	8.041813	3.350215	5.295570	3014.09775
SSA	3.5000216	0.700000	17.000	7.604074	7.807178	3.351558	5.286686	2999.54113
BOA	3.5258583	0.700000	17.000	7.300000	8.129062	3.418148	5.386728	3096.43410
HHO	3.5	0.700000	17.000	7.300000	7.875596	3.354856	5.289104	3000.73394
AOA	3.5051134	0.700000	17.000	7.300000	8.300000	3.465051	5.339268	3073.58747
MTV-WOA	3.5	0.700000	17.000	7.300000	7.715320	3.350215	5.286654	**2994.47107**

Best results among all algorithms are indicated in bold

usage of BS_TVP, LS_TVP, and GS_TVP on other algorithms, such as PSO and LSHADE-SPACMA.

Moreover, MTV-WOA showcased remarkable proficiency in handling equality and inequality constraints within different engineering design problems, as evidenced by Table 3, 4, 5, 6, 7 results. In engineering design contexts, MTV-WOA emerges as a valuable tool, streamlining design processes, reducing development time, and enhancing project efficiency. Moreover, the algorithm's risk mitigation capabilities make it an asset in navigating uncertain decision-making landscapes. With cross-industry applicability, managers across diverse sectors can leverage MTV-WOA to optimize processes and address industry-specific challenges. Implementing MTV-WOA can also elevate operational efficiency, allowing managers to fine-tune processes, optimize workflows, and achieve improved performance metrics. This adaptability extends to resource-intensive sectors, where the algorithm proves instrumental in optimizing resource allocation, contributing to overall cost-effectiveness in operational workflows.

As in all studies, the proposed algorithm has some limitations. The MTV-WOA proposed in this research is designed to solve single-objective and continuous optimization problems. It is acknowledged that there exists a notable gap in addressing multi-objective and discrete problems, prompting a recognition of the need for future research to extend the algorithm's applicability to these domains. The winner-based distribution policy, designed for the three TVPs employed in this study, may necessitate adaptation for handling new trial vectors in different problems. Furthermore, the MTV-WOA was not evaluated for large-scale global optimization (LSGO) problems, and its performance may be limited when the dimension is increased. To address this issue, it is essential to determine the size of the lifetime archive and establish a suitable policy for high-dimensional problems. Pre-experimental investigations are necessary to optimize these parameters and enhance the algorithm's performance in problems with increased problem dimensions.

8 Conclusion

This study addresses the limitations of stochastic algorithms, particularly swarm intelligence metaheuristic algorithms, in dealing with complex problems. It introduces the Multi-trial Vector-based Whale Optimization Algorithm (MTV-WOA) as an enhancement over the canonical Whale Optimization Algorithm (WOA). The conventional WOA exhibits challenges such as an imbalance between exploration and exploitation, leading to premature convergence. In response, the study employs the Multi-trial Vector (MTV) approach, incorporating three TVPs to replace the WOA search strategy. The MTV-WOA introduces three new strategies, BS_TVP, LS_TVP, and GS_TVP, to address diverse problems with distinct characteristics. Experimental validation using the CEC 2018 test suite demonstrates the superiority of MTV-WOA over three classes of existing optimization algorithms: recently published, well-established, and highly performing algorithms that are winners of CEC competitions in terms of exploration, exploitation, local optima avoidance, and convergence. The Friedman and Wilcoxon signed-rank tests establish the statistical significance of MTV-WOA's performance, affirming its efficacy in maintaining a balanced exploration–exploitation trade-off. Additionally, the study showcases MTV-WOA's practical applicability by addressing five engineering design problems, where it consistently outperforms alternative algorithms.

The MTV-WOA is designed for continuous problems with a single objective. In future study, discrete and multi-objective real-world issues can be addressed by modifying MTV-WOA to handle binary and multi-objective problems, depending on the problem's nature. It is also beneficial to try to tackle issues in other fields, such as disease diagnosis by feature selection, image processing, and community identification.

Appendix A

See Tables 8, 9, 10, 11, 12; Figs. 3, 4.

Table 8 The unimodal functions' average error

F	D		Alg. KH	Alg. GWO	Alg. MFO	Alg. WOA	Alg. SSA	Alg. HHO	Alg. BOA	Alg. AOA	Alg. MTV-WOA
F1	10		8.8828E+02	2.0334E+07	7.8026E+07	3.4908E+05	2.3885E+03	2.3357E+05	2.4796E+08	2.5055E+09	**6.3349E-02**
	30		2.1874E+04	8.8219E+08	6.8955E+09	2.2829E+06	6.9320E+03	8.2109E+06	1.1853E+10	4.0129E+10	**5.5226E-03**
	50		2.7044E+05	4.3655E+09	3.1958E+10	6.8040E+06	6.2256E+03	3.4717E+07	4.4742E+10	1.0163E+11	**2.6153E-01**
F3	10		5.8996E+02	7.4837E+02	5.3397E+03	2.1480E+02	**1.3059E-09**	8.2588E-01	2.0913E+02	3.0925E+03	2.8278E-06
	30		4.0298E+04	2.6236E+04	8.8266E+04	1.3475E+05	**4.0331E-08**	6.3908E+02	3.0697E+04	7.2851E+04	7.7724E-05
	50		1.1412E+05	7.7633E+04	1.9713E+05	7.1094E+04	**1.6485E-07**	2.6461E+03	2.1583E+05	1.7433E+05	5.0485E-04
Overall Rank	10	w/t/l	0/0/2	0/0/2	0/0/2	0/0/2	1/0/1	0/0/2	0/0/2	0/0/2	1/0/1
	30	w/t/l	0/0/2	0/0/2	0/0/2	0/0/2	1/0/1	0/0/2	0/0/2	0/0/2	1/0/1
	50	w/t/l	0/0/2	0/0/2	0/0/2	0/0/2	1/0/1	0/0/2	0/0/2	0/0/2	1/0/1

Best results among all algorithms are indicated in bold

Table 9 The multimodal functions' average error

F	D		Alg. KH	Alg. GWO	Alg. MFO	Alg. WOA	Alg. SSA	Alg. HHO	Alg. BOA	Alg. AOA	Alg. MTV-WOA
F4	10		6.0063E+00	1.2149E+01	5.9214E+00	3.1464E+01	3.7667E+00	1.7283E+01	2.9595E+01	1.0258E+02	**1.3075E-01**
	30		9.7472E+01	1.5290E+02	3.6285E+02	1.2508E+02	9.7071E+01	1.2277E+02	1.5792E+03	8.6134E+03	**7.0727E+01**
	50		1.7035E+02	3.8914E+02	3.3271E+03	2.7695E+02	1.1802E+02	2.1865E+02	5.9694E+03	2.4422E+04	**8.5538E+01**
F5	10		2.2735E+01	1.2906E+01	2.2687E+01	5.1795E+01	2.2181E+01	4.0434E+01	6.5337E+01	5.0016E+01	**8.0668E+00**
	30		1.3006E+02	9.8392E+01	1.8291E+02	2.7225E+02	1.3258E+02	2.0965E+02	3.3233E+02	3.1444E+02	**7.7470E+01**
	50		2.6732E+02	1.9006E+02	4.2470E+02	4.1031E+02	2.7472E+02	3.6043E+02	5.9792E+02	5.7180E+02	**1.6330E+02**
F6	10		3.9010E+00	4.8103E-01	5.3799E-01	2.8575E+01	7.8912E+00	2.3011E+01	1.3862E+01	3.5503E+01	**1.8199E-01**
	30		4.5233E+01	3.7354E+00	2.5397E+01	6.7463E+01	3.6927E+01	5.8973E+01	4.8259E+01	6.5827E+01	**5.4787E-02**
	50		5.0322E+01	1.1021E+01	4.8175E+01	7.8083E+01	4.6352E+01	6.9354E+01	6.8678E+01	8.4863E+01	**4.1208E-02**
F7	10		2.0100E+01	2.4629E+01	3.6945E+01	7.5273E+01	3.4854E+01	6.9121E+01	7.2820E+01	1.0283E+02	**2.3383E+01**
	30		1.3424E+02	1.5427E+02	2.9749E+02	5.2518E+02	1.6496E+02	5.0044E+02	5.0075E+02	5.8170E+02	**1.2304E+02**
	50		3.4887E+02	2.8392E+02	8.9914E+02	9.6230E+02	3.3374E+02	1.0089E+03	9.5368E+02	1.1166E+03	**2.5393E+02**
F8	10		1.7163E+01	1.1286E+01	2.0848E+01	3.6691E+01	1.9551E+01	2.6533E+01	4.0891E+01	2.7043E+01	**1.0561E+01**
	30		1.1210E+02	8.5054E+01	1.7552E+02	2.0096E+02	1.4178E+02	1.5657E+02	2.8681E+02	2.2355E+02	**7.2436E+01**
	50		2.6584E+02	1.8796E+02	4.3178E+02	4.3682E+02	2.7510E+02	3.4934E+02	6.2159E+02	6.1927E+02	**1.6091E+02**
F9	10		5.2120E-03	4.5125E-01	2.2572E+01	4.2953E+02	2.6120E+00	4.4480E+02	2.2035E+01	4.4789E+02	**1.4452E-01**
	30		2.1477E+03	6.9167E+02	4.8494E+03	6.6729E+03	2.7090E+03	4.9406E+03	5.1511E+03	4.7583E+03	**7.2685E-01**
	50		8.8317E+03	3.3173E+03	1.3801E+04	1.9254E+04	1.0157E+04	1.5484E+04	2.4289E+04	2.1180E+04	**4.3486E+01**
F10	10		1.0131E+03	4.8632E+02	7.4973E+02	9.8924E+02	6.5908E+02	1.0509E+03	1.3559E+03	1.0435E+03	**4.3022E+02**
	30		3.9477E+03	**2.8045E+03**	4.2244E+03	5.3774E+03	3.7869E+03	4.2207E+03	7.1413E+03	5.2395E+03	4.1393E+03
	50		7.3036E+03	**5.5198E+03**	6.7742E+03	8.7723E+03	6.7938E+03	6.9154E+03	1.3388E+04	1.1497E+04	8.3187E+03
Overall Rank	10	w/t/l	1/0/6	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	6/0/1
	30	w/t/l	0/0/7	1/0/6	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	6/0/1
	50	w/t/l	0/0/7	1/0/6	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	0/0/7	6/0/1

Best results among all algorithms are indicated in bold

Table 10 The hybrid functions' average error

F	D		Alg. KH	Alg. GWO	Alg. MFO	Alg. WOA	Alg. SSA	Alg. HHO	Alg. BOA	Alg. AOA	Alg. MTV-WOA
F11	10		2.8294E+01	2.1170E+01	3.3879E+01	8.3171E+01	6.5405E+01	5.4816E+01	1.1229E+02	5.9905E+01	**4.1487E+00**
	30		5.1975E+02	4.3260E+02	1.7479E+03	3.6681E+02	1.8621E+02	1.4996E+02	1.5337E+03	1.9833E+03	**5.0187E+01**
	50		3.7039E+03	1.9400E+03	7.1346E+03	5.0877E+02	2.7644E+02	3.3795E+02	1.5620E+03	1.5092E+04	**1.2718E+02**
F12	10		1.4852E+06	5.5111E+05	2.0350E+06	3.7211E+06	1.6943E+06	1.2153E+06	2.8599E+06	1.5369E+06	**1.1532E+02**
	30		3.1075E+06	3.8433E+07	2.4446E+08	3.7742E+07	3.3686E+06	7.9605E+06	6.0509E+08	7.8447E+09	**1.1966E+03**
	50		1.0268E+07	2.9199E+08	3.5160E+09	2.0148E+08	1.6697E+07	4.3367E+07	9.4035E+09	5.4454E+10	**3.1582E+03**
F13	10		1.0070E+04	8.3472E+03	9.3832E+03	1.2535E+04	1.7649E+04	1.1843E+04	2.4036E+04	1.0942E+04	**8.0730E+00**
	30		4.1755E+04	9.2195E+04	4.0913E+06	1.5901E+05	1.0542E+05	1.9685E+05	7.0337E+07	3.1890E+04	**7.6813E+01**
	50		4.6996E+04	1.1417E+08	3.4745E+08	1.8967E+05	1.0815E+05	1.3040E+06	8.9008E+08	4.2858E+09	**2.2047E+02**
F14	10		6.4657E+02	4.1839E+02	9.4319E+02	1.6319E+02	8.8752E+01	1.1252E+02	4.9201E+02	9.6044E+03	**6.3600E+00**
	30		2.5784E+05	2.1312E+05	1.6478E+05	8.7208E+05	4.9758E+03	2.2891E+04	9.5668E+04	5.1598E+04	**4.1982E+01**
	50		5.0509E+05	3.6925E+05	1.0022E+06	4.6928E+05	4.2261E+04	2.1655E+05	4.6192E+05	3.1976E+05	**7.8229E+01**
F15	10		3.4204E+03	1.9271E+03	2.5376E+03	2.6962E+03	6.8142E+02	6.6312E+02	1.6240E+03	8.2839E+03	**1.7435E+00**
	30		2.0051E+04	1.8474E+05	4.4100E+04	8.7669E+04	5.3641E+04	5.5106E+04	1.3410E+06	2.2583E+04	**2.7838E+01**
	50		1.9487E+04	9.5200E+06	2.8132E+07	1.0220E+05	5.3395E+04	1.6151E+05	3.2275E+07	3.1760E+04	**1.0209E+02**
F16	10		3.0134E+02	9.7698E+01	9.0706E+01	1.8929E+02	8.8330E+01	2.4975E+02	1.7457E+02	3.8357E+02	**3.2250E+00**
	30		1.1444E+03	7.3557E+02	1.2834E+03	1.8296E+03	8.4340E+02	1.4767E+03	2.7904E+03	2.3154E+03	**5.0636E+02**
	50		1.7043E+03	**1.1648E+03**	2.6525E+03	3.2622E+03	1.5580E+03	2.2948E+03	4.8317E+03	4.4119E+03	1.2962E+03
F17	10		6.8011E+01	3.9850E+01	6.1100E+01	9.4911E+01	5.6778E+01	6.6308E+01	7.6950E+01	1.6872E+02	**2.1762E+01**
	30		5.2483E+02	1.9731E+02	6.8635E+02	8.2320E+02	3.9109E+02	8.1966E+02	1.1080E+03	9.1089E+02	**1.2778E+02**
	50		1.5803E+03	1.0027E+03	2.2440E+03	2.2269E+03	1.4093E+03	1.8518E+03	2.7596E+03	2.2653E+03	**7.5240E+02**
F18	10		9.7159E+03	1.8019E+04	2.1570E+04	1.0826E+04	1.5004E+04	1.2478E+04	1.6910E+05	1.1482E+04	**5.0234E+00**
	30		3.1180E+05	4.9367E+05	3.6481E+06	2.0594E+06	2.2859E+05	7.6568E+05	1.1046E+06	4.6083E+05	**3.3149E+01**
	50		2.4602E+06	1.7119E+06	1.0956E+07	4.5270E+06	3.6924E+05	2.2727E+06	1.5759E+06	2.1734E+07	**5.9225E+01**
F19	10		2.8041E+03	4.0596E+03	1.4452E+04	2.5756E+04	1.7311E+02	7.0888E+03	1.0011E+04	2.8637E+04	**1.7015E+00**
	30		5.5602E+04	2.8040E+05	4.1704E+06	2.9504E+06	4.9892E+05	2.5955E+05	2.7794E+06	1.0655E+06	**1.9554E+01**
	50		1.7780E+05	2.5297E+06	1.0124E+07	2.4686E+06	7.0703E+05	3.6583E+05	1.5345E+07	4.6484E+05	**4.4346E+01**
F20	10		1.2427E+02	6.4831E+01	4.7565E+01	1.3996E+02	8.2026E+01	1.2424E+02	1.1107E+02	1.0558E+02	**1.5589E+01**
	30		6.3217E+02	3.0442E+02	5.7465E+02	7.6804E+02	4.5434E+02	5.9459E+02	7.6354E+02	7.4069E+02	**1.2100E+02**
	50		1.3383E+03	7.1396E+02	1.4478E+03	1.5204E+03	9.9376E+02	1.3637E+03	1.8726E+03	1.3811E+03	**5.3929E+02**
Overall Rank	10	w/t/l	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	10/0/0
	30	w/t/l	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	10/0/0
	50	w/t/l	0/0/10	1/0/9	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	9/0/1

Best results among all algorithms are indicated in bold

Table 11 The composition functions' average error

F	D		Alg. KH	Alg. GWO	Alg. MFO	Alg. WOA	Alg. SSA	Alg. HHO	Alg. BOA	Alg. AOA	Alg. MTV-WOA
F21	10		1.1647E+02	1.8648E+02	2.0367E+02	1.7375E+02	1.4322E+02	2.0067E+02	**1.0373E+02**	2.0445E+02	1.4504E+02
	30		3.0806E+02	2.7439E+02	3.7965E+02	4.6149E+02	2.9036E+02	4.2877E+02	**1.6107E+02**	5.0294E+02	2.7757E+02
	50		4.3958E+02	3.7657E+02	5.9997E+02	7.6597E+02	4.6011E+02	6.7571E+02	4.0665E+02	9.1365E+02	**3.7389E+02**
F22	10		1.0019E+02	1.0577E+02	1.0238E+02	1.1533E+02	1.0354E+02	1.1485E+02	1.0618E+02	3.6711E+02	**4.9060E+01**
	30		8.6476E+02	2.2620E+03	4.3371E+03	3.9653E+03	9.7092E+02	3.6642E+03	5.6186E+02	5.7881E+03	**3.0149E+02**
	50		7.9684E+03	**6.4568E+03**	8.2570E+03	9.6743E+03	7.2278E+03	8.3919E+03	1.0999E+04	1.2793E+04	8.6419E+03
F23	10		3.3308E+02	3.1515E+02	3.2600E+02	3.5025E+02	3.2105E+02	3.5605E+02	**2.7338E+02**	3.9761E+02	3.0722E+02
	30		6.0013E+02	4.4197E+02	5.1769E+02	7.1771E+02	4.4483E+02	7.7858E+02	6.3987E+02	1.0636E+03	**4.2736E+02**
	50		1.0551E+03	6.1472E+02	8.0479E+02	1.2626E+03	6.8450E+02	1.2824E+03	1.2087E+03	1.9508E+03	**6.1600E+02**
F24	10		3.1127E+02	3.3473E+02	3.5759E+02	3.5455E+02	3.1937E+02	3.8473E+02	**1.3402E+02**	4.2477E+02	3.1359E+02
	30		7.1088E+02	5.1114E+02	5.7323E+02	7.7881E+02	5.2269E+02	9.2091E+02	5.4248E+02	1.2541E+03	**4.9134E+02**
	50		1.2541E+03	6.8986E+02	8.0369E+02	1.2201E+03	7.1071E+02	1.6236E+03	1.6475E+03	2.3400E+03	**6.6237E+02**
F25	10		4.2410E+02	4.3433E+02	4.3924E+02	4.1910E+02	4.1961E+02	4.0664E+02	3.9970E+02	5.3127E+02	**3.8883E+02**
	30		4.2523E+02	4.6179E+02	6.0140E+02	4.5467E+02	4.0227E+02	4.0615E+02	7.3232E+02	1.8773E+03	**3.8664E+02**
	50		5.8724E+02	8.9885E+02	2.5965E+03	6.5220E+02	5.3347E+02	6.2498E+02	3.9644E+03	1.1017E+04	**4.9130E+02**
F26	10		3.8623E+02	3.5615E+02	3.8845E+02	6.6093E+02	2.8500E+02	7.1571E+02	3.1512E+02	9.7592E+02	**2.8221E+02**
	30		3.2546E+03	1.9393E+03	2.8953E+03	5.2407E+03	1.8031E+03	4.0374E+03	**1.4963E+03**	7.0728E+03	1.8546E+03
	50		7.4365E+03	3.2334E+03	5.5408E+03	1.1009E+04	**1.2032E+03**	7.2522E+03	5.7094E+03	1.3062E+04	2.7951E+03
F27	10		4.1975E+02	3.9365E+02	3.9221E+02	4.1864E+02	3.9131E+02	4.3197E+02	4.0218E+02	4.9197E+02	**3.9056E+02**
	30		6.9887E+02	5.3956E+02	5.3848E+02	6.7390E+02	5.2944E+02	5.9891E+02	6.0102E+02	1.5682E+03	**5.2079E+02**
	50		1.6560E+03	8.0556E+02	8.2975E+02	1.6528E+03	7.0895E+02	1.2178E+03	9.7957E+02	3.5778E+03	**5.9911E+02**
F28	10		3.9518E+02	5.7778E+02	4.9047E+02	5.8402E+02	4.0322E+02	5.1405E+02	3.6791E+02	7.3848E+02	**2.8077E+02**
	30		4.3975E+02	5.5750E+02	1.3499E+03	4.9031E+02	4.1753E+02	4.5500E+02	1.1076E+03	2.9936E+03	**3.1012E+02**
	50		5.3316E+02	1.0730E+03	5.1319E+03	6.3158E+02	5.0077E+02	5.4970E+02	3.6051E+03	7.9944E+03	**4.5885E+02**
F29	10		3.4510E+02	2.8435E+02	2.8763E+02	4.2334E+02	2.7827E+02	4.0927E+02	3.4826E+02	4.1132E+02	**2.5370E+02**
	30		1.1524E+03	7.4063E+02	9.6392E+02	1.8523E+03	1.0130E+03	1.4100E+03	2.1697E+03	3.0675E+03	**6.5621E+02**
	50		2.4004E+03	1.3842E+03	2.2084E+03	4.3782E+03	1.8193E+03	2.1142E+03	4.9410E+03	1.1625E+04	**9.2126E+02**
F30	10		6.3885E+05	8.4424E+05	4.9646E+05	8.4594E+05	9.9043E+04	3.7895E+05	3.2699E+05	5.1769E+06	**4.9674E+02**
	30		2.1421E+06	5.1737E+06	2.5012E+05	1.0055E+07	1.7900E+06	8.8139E+05	1.6499E+07	2.8144E+08	**2.1610E+03**
	50		3.9710E+07	7.1106E+07	6.4895E+07	9.8372E+07	2.8292E+07	1.2956E+07	2.0233E+08	3.2535E+08	**7.0599E+05**
Overall Rank	10	w/t/l	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	3/0/7	0/0/10	7/0/3
	30	w/t/l	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	0/0/10	2/0/8	0/0/10	8/0/2
	50	w/t/l	0/0/10	1/0/9	0/0/10	0/0/10	1/0/9	0/0/10	0/0/10	0/0/10	8/0/2

Best results among all algorithms are indicated in bold

Table 12 The Friedman test results

Alg.	D	F1	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17
KH	10	2.45	6.5	4.35	4	3.9	2.05	3.75	1.95	6.1	4.05	5.85	5.1	5.8	5.85	7.3	5.5
	30	2.85	5.95	2.95	3.50	5.00	2.50	3.30	3.35	3.85	5.75	2.75	3.30	6.90	2.95	4.40	4.45
	50	3	6	3.05	3.5	4.2	3.3	3.7	3.3	4.15	7.25	2.3	2.2	6.45	2.3	3.4	4.1
GWO	10	5.6	5.9	5.7	2.15	2.25	2.5	2.25	4.1	2.3	3.45	3.95	4.6	3.4	5.1	4.05	3
	30	6.05	4.70	5.45	2.15	2.00	2.75	2.10	2.30	1.45	4.70	5.45	5.30	5.20	5.40	2.65	1.80
	50	6.05	4.95	5.85	1.95	2	2.15	1.9	2.1	1.4	5.65	5.45	6.15	5.45	6.05	2.05	1.95
MFO	10	4.45	5.55	4.25	4.25	2.2	4.35	4.7	2.65	4.4	3.5	4.35	4.2	6.7	5.7	3.4	4.65
	30	7.20	7.55	6.35	5.10	3.45	5.00	6.00	6.35	4.65	6.85	5.90	4.75	5.35	5.25	4.90	5.45
	50	7.25	7.5	7.1	6.45	3.85	6.3	6.3	5.4	3.5	7.25	6.95	6.35	5.6	6.35	5.7	6.9
WOA	10	5.5	5.6	6.15	7.2	7.95	7	7.6	7.9	5.8	6.5	6.6	5.1	5.15	5.9	5.7	6.75
	30	4.00	8.30	4.65	7.15	8.15	7.20	6.75	8.10	7.00	5.10	5.95	6.05	8.60	6.30	6.50	6.55
	50	4	4.75	4.85	6.1	7.8	6.85	6.4	7.1	6	3.9	5.4	3.8	7.2	5.65	6.85	6.4
SSA	10	2.9	1.4	2.75	3.9	5	3.85	4.1	3.35	3.25	5.9	5.95	6.4	3.4	3.7	3.7	4.15
	30	2.15	1.00	2.75	3.50	4.55	3.35	4.35	4.05	3.55	3.00	2.90	5.75	2.70	5.75	2.70	3.50
	50	2	1	2.05	3.55	4.15	3.25	3.5	3.85	3.35	2.35	2.8	3.4	2.4	4.6	3.1	3.6
HHO	10	6.2	3.4	4.8	6.4	7	6.75	6.05	8.05	6.4	5.95	4.7	5.5	4.35	3.75	6	5.2
	30	5.00	3.00	4.35	5.60	7.30	7.05	5.05	6.55	4.55	2.35	4.25	7.05	3.55	6.05	5.75	6.40
	50	5	3	3.75	5.25	6.85	7.25	4.95	6	3.6	2.8	4.15	5.6	4.75	6.8	5.2	5.3
BOA	10	7.9	5.85	7.35	8.55	5.9	7.05	8.4	5.85	8.3	8.25	7.25	7.8	6.9	5.55	5.7	6.3
	30	7.75	4.90	7.95	8.45	5.50	7.10	8.90	6.90	9.00	8.05	7.80	8.95	6.60	8.90	8.80	8.30
	50	7.7	8.35	7.9	8.8	6.65	6.35	8.45	8.4	9	5.85	7.95	8	6.9	8.65	8.6	8.4
AOA	10	9	8.4	8.6	7.3	8.4	8.9	6.15	8	6.5	6.25	5.35	5.3	8.3	8.45	8.05	8.3
	30	9.00	7.60	9.00	8.15	8.05	8.20	7.25	6.40	6.80	8.20	9.00	2.85	5.10	3.40	7.90	7.15
	50	9	7.45	9	8.2	8.5	8.25	8.5	7.85	7.95	8.95	9	8.5	5.25	3.6	8.1	6.9
MTV-WOA	10	1	2.4	1.05	1.25	2.4	2.55	2	3.15	1.95	1.15	1	1	1	1	1.1	1.15
	30	1.00	2.00	1.55	1.40	1.00	1.85	1.30	1.00	4.15	1.00	1.00	1.00	1.00	1.00	1.40	1.40
	50	1	2	1.45	1.2	1	1.3	1.3	1	6.05	1	1	1	1	1	2	1.45

Alg.	D	F18	F19	F20	F21	F22	F23	F24	F25	F26	F27	F28	F29	F30	Avg. rank	Overall ranl
KH	10	4.4	4.35	6.7	3.1	3.45	5.15	4.35	4.25	4.3	6.7	3.3	5.8	5.55	4.69	5
	30	3.55	2.7	5.55	4.5	3.2	5.4	6.1	4.05	5.4	7.4	3.2	4.7	4.45	4.27	4
	50	5.85	2.95	5.25	3.95	3.7	5.35	5.55	3.05	6.25	7.55	3.2	5.2	4.75	4.30	4
GWO	10	5.9	4.05	4	5.5	4.65	2.65	3.35	5.85	4.3	3.55	7.1	3.3	5.85	4.15	3
	30	4.55	4.15	2.45	2.8	5.05	2	2.45	5.5	3.55	3.2	5.95	2.2	6	3.77	3
	50	4.45	5.3	2.1	2.2	1.85	1.65	2.1	5.8	2.8	3.25	6	2.2	5.85	3.68	3
MFO	10	6.25	5.85	2.95	6.65	3.95	4.5	5.95	6.4	5.5	2.75	5.6	3.5	5.5	4.64	4
	30	6.55	3.35	5.35	6.35	6.75	4.15	4.45	6.45	5.3	3	7.4	3.45	2.25	5.34	5
	50	6.25	4.4	6.1	6.05	4.6	3.9	3.85	7.15	4.4	3.7	8	4.6	3.65	5.70	6
WOA	10	4.55	6.85	6.8	6.3	6.85	6.85	6.65	6.25	6.9	6.6	6.75	7.4	5.55	6.44	8
	30	7.60	8.15	7.2	7.85	6.8	7.1	6.75	5.35	7.65	6.5	4.85	7.2	6.65	6.76	8
	50	6.95	7.05	6.65	7.85	6.25	6.95	5.6	4.75	8	7.25	4.7	7.2	6.7	6.17	7
SSA	10	4.7	3	4.7	3.5	4.35	3.8	4.35	4.1	2.05	1.9	3.25	3.1	3.05	3.78	2
	30	3.20	5.7	3.7	3.85	2.95	2.45	3.2	2.9	3.1	2.35	2.5	3.75	4.75	3.45	2
	50	2.3	5.85	3.75	4.05	2.9	2.75	2.4	2	1.55	2.3	2.35	3.65	3.85	3.06	2
HHO	10	4.65	5.95	6.6	6.45	6.7	7.3	7.5	4.6	7.25	7.5	5.75	7.2	4.8	5.96	6
	30	6.05	4.65	5.25	7	6.15	7.4	7.85	3	6.2	5.9	3.6	5.65	3.75	5.39	6
	50	5.5	4.3	5.7	6.9	4.7	7.1	7.3	4.25	6.1	6	3.75	4.35	2.5	5.13	5
BOA	10	8.8	4.35	6.7	3.1	3.45	5.15	4.35	4.25	4.3	6.7	3.3	5.8	5.55	6.09	7
	30	7.75	2.7	5.55	4.5	3.2	5.4	6.1	4.05	5.4	7.4	3.2	4.7	4.45	6.92	7
	50	4.4	2.95	5.25	3.95	3.7	5.35	5.55	3.05	6.25	7.55	3.2	5.2	4.75	7.21	8
AOA	10	4.75	4.05	4	5.5	4.65	2.65	3.35	5.85	4.3	3.55	7.1	3.3	5.85	7.60	9
	30	4.75	4.15	2.45	2.8	5.05	2	2.45	5.5	3.55	3.2	5.95	2.2	6	7.60	9
	50	8.3	5.3	2.1	2.2	1.85	1.65	2.1	5.8	2.8	3.25	6	2.2	5.85	8.04	9

Table 12 (continued)

Alg.	D	F18	F19	F20	F21	F22	F23	F24	F25	F26	F27	F28	F29	F30	Avg. rank	Overall ranl
MTV-WOA	10	1	5.85	2.95	6.65	3.95	4.5	5.95	6.4	5.5	2.75	5.6	3.5	5.5	1.66	**1**
	30	1.00	3.35	5.35	6.35	6.75	4.15	4.45	6.45	5.3	3	7.4	3.45	2.25	1.51	**1**
	50	1	4.4	6.1	6.05	4.6	3.9	3.85	7.15	4.4	3.7	8	4.6	3.65	1.60	**1**

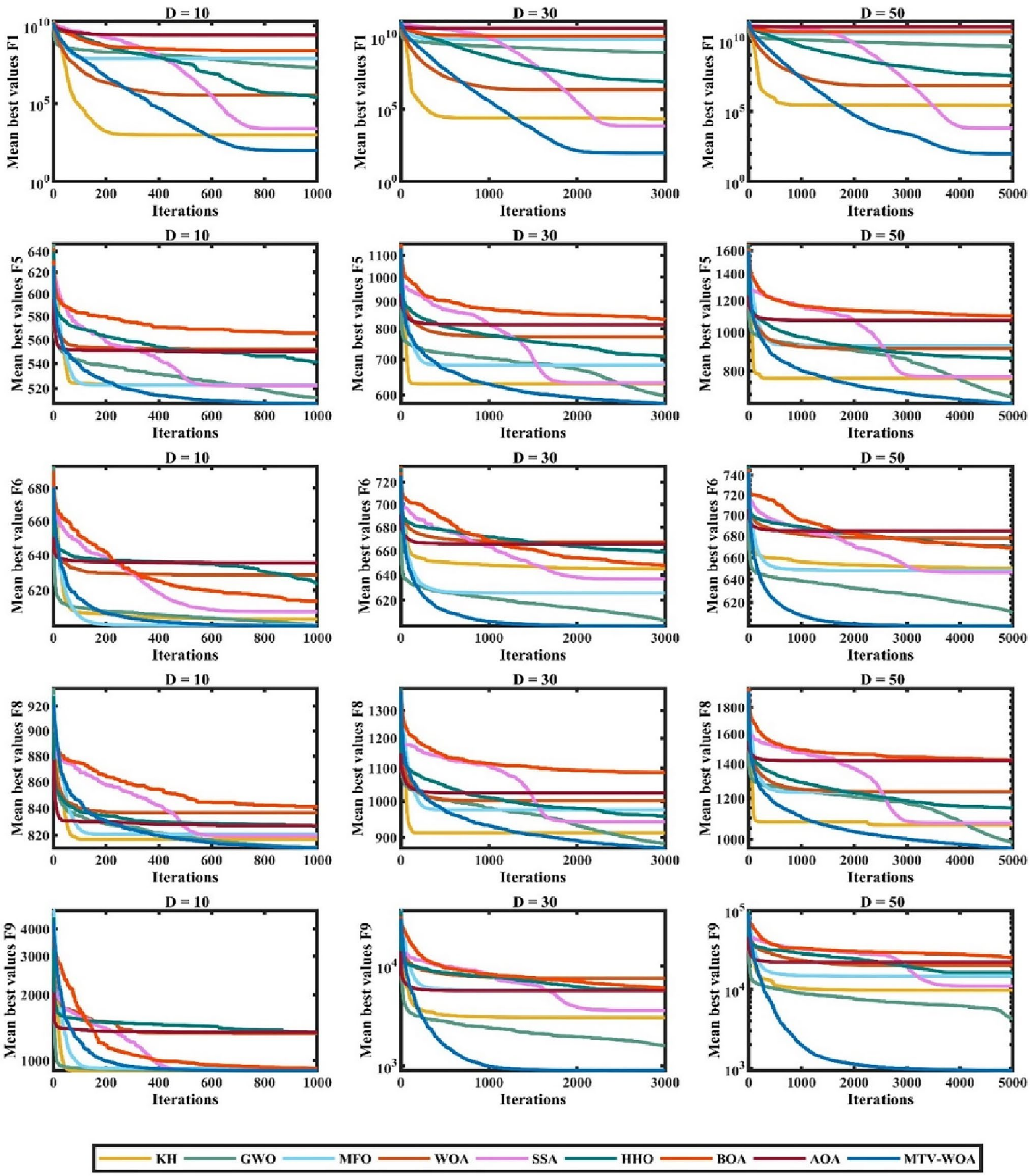


Fig. 3 Convergence curves for some unimodal and multimodal functions

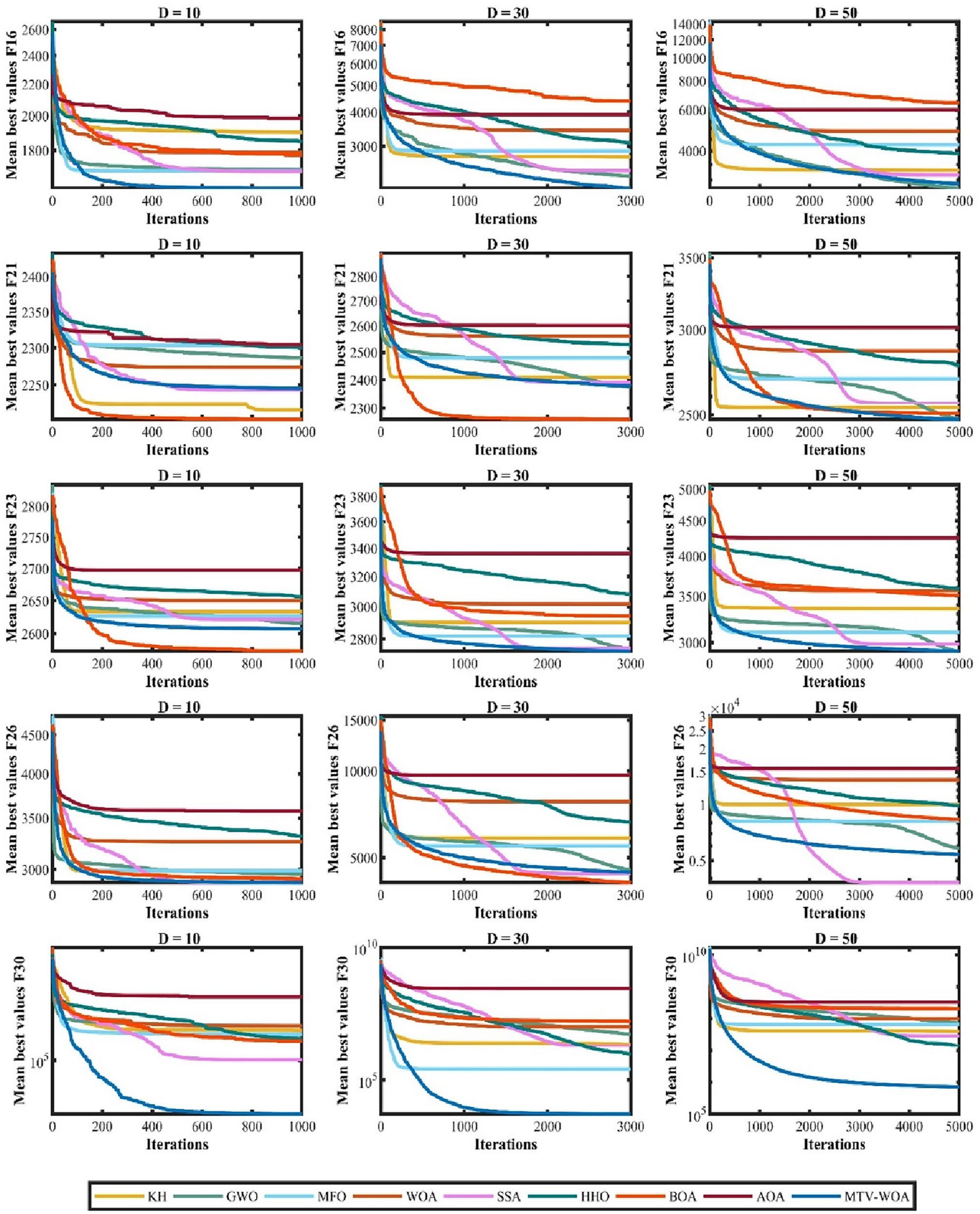


Fig. 4 Convergence curves for some hybrid and composition functions

Appendix B

See Figs. 5, 6; Tables 13, 14, 15.

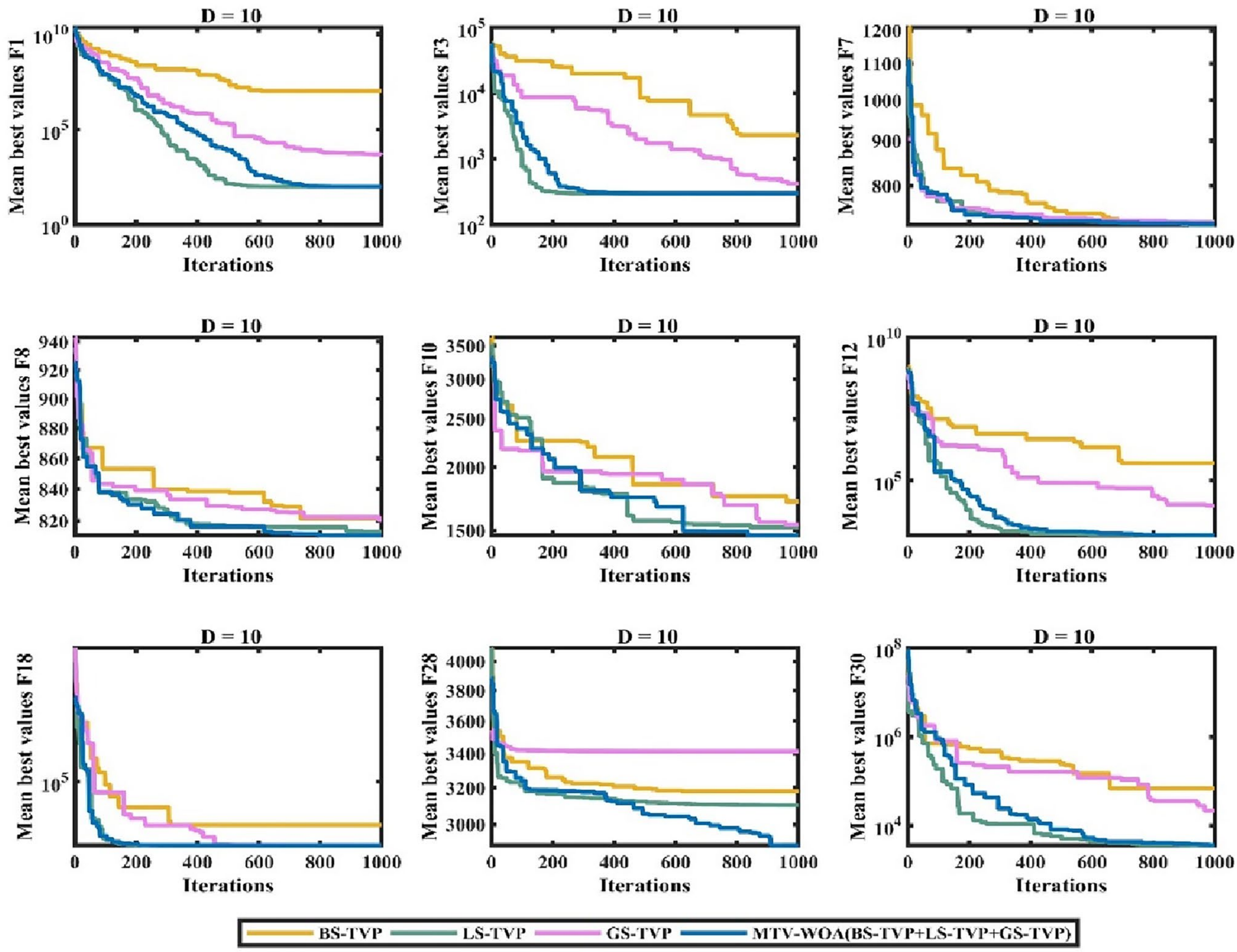


Fig. 5 The impact analysis of using the proposed TVPs

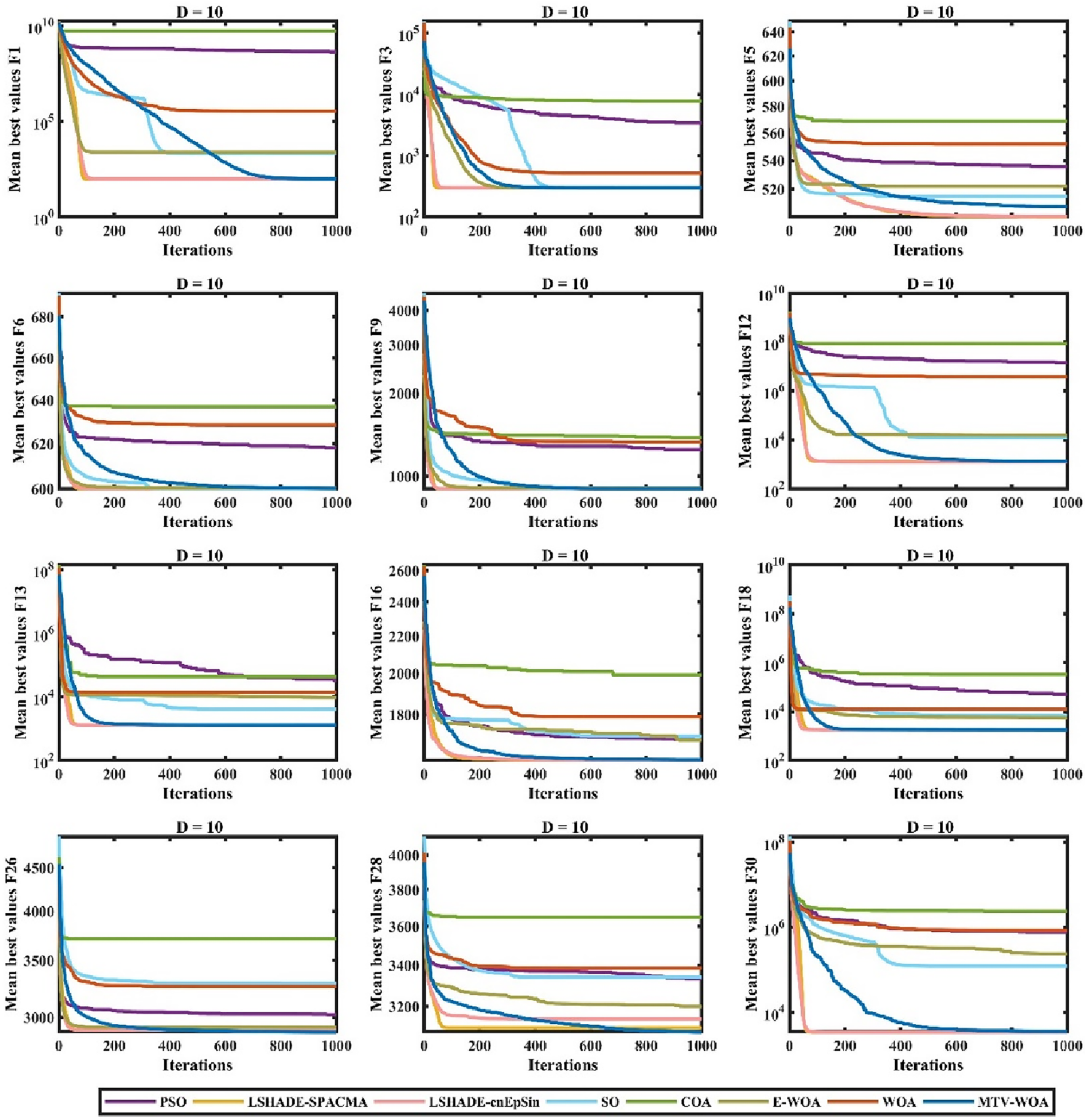


Fig. 6 Convergence comparison of MTV-WOA with flagship, recent, and WOA variant algorithms

Table 13 The comparison of MTV-WOA with well-stablished, recent, and WOA variant algorithms

F	Alg. PSO	Alg. LSHADE-SPACMA	Alg. LSHADE – cnEp-Sin	Alg. SO	Alg. COA	Alg. E-WOA	Alg. WOA	Alg. MTV-WOA
F1	4.4224E+08	0.0000E+00	0.0000E+00	2.0241E+03	5.4568E+09	2.3740E+03	3.4908E+05	6.3349E-02
F3	3.0775E+03	0.0000E+00	0.0000E+00	7.0486E-03	7.4050E+03	5.4001E-14	2.1480E+02	2.8278E-06
F4	3.9140E+01	0.0000E+00	0.0000E+00	3.0997E+00	4.3521E+02	1.7239E+00	3.1464E+01	1.3075E-01
F5	3.5604E+01	1.2088E+00	1.6322E+00	1.4932E+01	6.8975E+01	2.1641E+01	5.1795E+01	8.0668E+00
F6	1.8216E+01	0.0000E+00	0.0000E+00	4.2757E-02	3.7022E+01	2.8874E-01	2.8575E+01	1.8199E-01
F7	1.1527E+02	1.1111E+01	1.1698E+01	2.7238E+01	8.2306E+01	3.3476E+01	7.5273E+01	2.3383E+01
F8	4.7964E+01	7.0721E-01	1.8685E+00	1.4079E+01	4.2276E+01	2.1640E+01	3.6691E+01	1.0561E+01
F9	3.4860E+02	0.0000E+00	0.0000E+00	2.6614E+00	4.8198E+02	5.5407E+00	4.2953E+02	1.4452E-01
F10	1.2032E+03	5.0832E+00	9.2189E+00	5.2960E+02	1.4048E+03	5.4098E+02	9.8924E+02	4.3022E+02
F11	1.3798E+02	0.0000E+00	0.0000E+00	1.1676E+01	3.8259E+02	1.4543E+01	8.3171E+01	4.1487E+00
F12	1.4391E+07	1.0200E+02	3.0013E+01	1.1028E+04	8.5496E+07	1.4299E+04	3.7211E+06	1.1532E+02
F13	3.1259E+04	2.4529E+00	4.2083E+00	2.7952E+03	4.1749E+04	8.3379E+03	1.2535E+04	8.0730E+00
F14	1.1781E+02	0.0000E+00	3.9448E-04	1.1096E+02	1.2682E+02	5.5201E+01	1.6319E+02	6.3600E+00
F15	9.8117E+02	3.4751E-01	1.0804E-01	2.7290E+02	3.7018E+03	2.5516E+01	2.6962E+03	1.7435E+00
F16	8.6093E+01	5.7989E-01	8.3820E-01	1.0218E+02	3.9174E+02	8.6024E+01	1.8929E+02	3.2250E+00
F17	9.0496E+01	1.8265E-01	5.3922E-01	5.1639E+01	7.2665E+01	4.3098E+01	9.4911E+01	2.1762E+01
F18	4.8868E+04	4.4659E-01	1.9594E-01	5.1035E+03	3.2572E+05	3.6984E+03	1.0826E+04	5.0234E+00
F19	4.3338E+02	4.8829E-02	1.7289E-02	6.5010E+02	3.1624E+03	2.6790E+01	2.5756E+04	1.7015E+00
F20	8.8842E+01	9.3652E-02	2.0584E-01	4.7299E+01	1.7210E+02	2.0649E+01	1.3996E+02	1.5589E+01
F21	1.9905E+02	1.0000E+02	1.3061E+02	2.1560E+02	2.0872E+02	1.0041E+02	1.7375E+02	1.4504E+02
F22	1.8207E+02	1.0000E+02	1.0000E+02	1.0168E+02	5.1722E+02	1.0209E+02	1.1533E+02	4.9060E+01
F23	3.2982E+02	3.0067E+02	3.0116E+02	3.1690E+02	3.9515E+02	3.2174E+02	3.5025E+02	3.0722E+02
F24	3.6091E+02	2.5563E+02	2.7188E+02	3.3871E+02	3.9419E+02	2.1233E+02	3.5455E+02	3.1359E+02
F25	4.6689E+02	4.1840E+02	4.1611E+02	4.2835E+02	7.2501E+02	4.2667E+02	4.1910E+02	3.8883E+02
F26	4.2237E+02	3.0000E+02	3.0000E+02	6.8831E+02	1.1089E+03	3.2017E+02	6.6093E+02	2.8221E+02
F27	4.1132E+02	3.8952E+02	3.8878E+02	3.9959E+02	4.5248E+02	3.9681E+02	4.1864E+02	3.9056E+02
F28	5.3254E+02	3.0000E+02	3.4061E+02	5.3939E+02	8.4818E+02	3.9668E+02	5.8402E+02	2.8077E+02
F29	2.9916E+02	2.3572E+02	2.3361E+02	2.7189E+02	4.3012E+02	3.0173E+02	4.2334E+02	2.5370E+02
F30	7.6090E+05	4.0659E+02	4.0412E+02	1.1937E+05	2.3849E+06	2.3389E+05	8.4594E+05	4.9674E+02
Friedman rank	7	1	2	5	8	4	6	3

Table 14 Results of Wilcoxon's test on D = 10

MTV-WOA vs	R^+	R^-	p-value	$\alpha = 0.05$
PSO	435	0	2.5631E-06	Yes
LSHADE_SPACMA	92	343	6.6534E-03	Yes
LSHADE_cnEpSin	94	341	7.5746E-03	Yes
SO	433	2	3.1652E-06	Yes
COA	435	0	2.5631E-06	Yes
E-WOA	395	40	1.2398E-04	Yes
WOA	435	0	2.5631E-06	Yes

Table 15 The results of using proposed TVPs for improving PSO and LSHADE-SPACMA algorithms

F	Alg. PSO	Alg. Adapted-PSO	Alg. LSHADE-SPACMA	Alg. Adapted-LSHADE-SPACMA
F1	4.4224E+08	**3.3207E+01**	**0.0000E+00**	**0.0000E+00**
F3	3.0775E+03	**2.8003E-03**	**0.0000E+00**	**0.0000E+00**
F4	3.9140E+01	**7.0774E-01**	**0.0000E+00**	**0.0000E+00**
F5	3.5604E+01	**9.7764E+00**	**1.2088E+00**	2.6376E+00
F6	1.8216E+01	**1.4046E-01**	**0.0000E+00**	**0.0000E+00**
F7	1.1527E+02	**2.6463E+01**	1.1111E+01	**1.0986E+01**
F8	4.7964E+01	**1.1972E+01**	**7.0721E-01**	2.7822E+00
F9	3.4860E+02	**1.9261E-01**	**0.0000E+00**	**0.0000E+00**
F10	1.2032E+03	**4.6846E+02**	**5.0832E+00**	3.9631E+01
F11	1.3798E+02	**5.3970E+00**	**0.0000E+00**	**0.0000E+00**
F12	1.4391E+07	**2.2300E+02**	1.0200E+02	**9.6750E+01**
F13	3.1259E+04	**1.1249E+01**	2.4529E+00	**1.3955E+00**
F14	1.1781E+02	**8.5272E+00**	**0.0000E+00**	5.7333E-01
F15	9.8117E+02	**2.2870E+00**	3.4751E-01	**2.8087E-01**
F16	8.6093E+01	**3.9902E+00**	5.7989E-01	**4.7542E-01**
F17	9.0496E+01	**2.3996E+01**	1.8265E-01	**1.7779E-01**
F18	4.8868E+04	**8.2109E+00**	4.4659E-01	**3.8985E-01**
F19	4.3338E+02	**2.0126E+00**	4.8829E-02	**4.5107E-02**
F20	8.8842E+01	**1.9011E+01**	9.3652E-02	**1.5609E-02**
F21	1.9905E+02	**1.0010E+02**	**1.0000E+02**	**1.0000E+02**
F22	1.8207E+02	**8.2793E+01**	1.0000E+02	**4.4892E+01**
F23	3.2982E+02	**3.0754E+02**	**3.0067E+02**	3.0412E+02
F24	3.6091E+02	**3.1481E+02**	2.5563E+02	**1.3652E+02**
F25	4.6689E+02	**3.9833E+02**	4.1840E+02	**4.0025E+02**
F26	4.2237E+02	**3.0004E+02**	**3.0000E+02**	**3.0000E+02**
F27	4.1132E+02	**3.9033E+02**	**3.8952E+02**	3.9068E+02
F28	5.3254E+02	**3.0288E+02**	**3.0000E+02**	**3.0000E+02**
F29	2.9916E+02	**2.5373E+02**	**2.3572E+02**	2.4120E+02
F30	7.6090E+05	**7.1209E+02**	4.0659E+02	**4.0252E+02**

Best results among all algorithms are indicated in bold

Appendix C

C.1. Pressure Vessel Design Problem

The major aim of this problem, represented in Fig. 7, is optimizing the cost of material, forming, and welding a vessel. The problem has four variables T_s, T_h, R, and L. The mathematical representation of this problem is provided in Eq. (24) (Figs. 7, 8, 9, 10, 11).

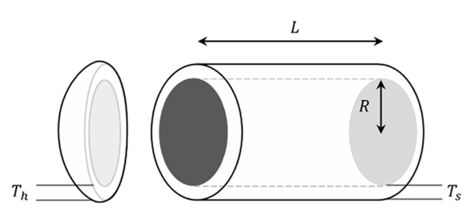


Fig. 7 Pressure vessel design

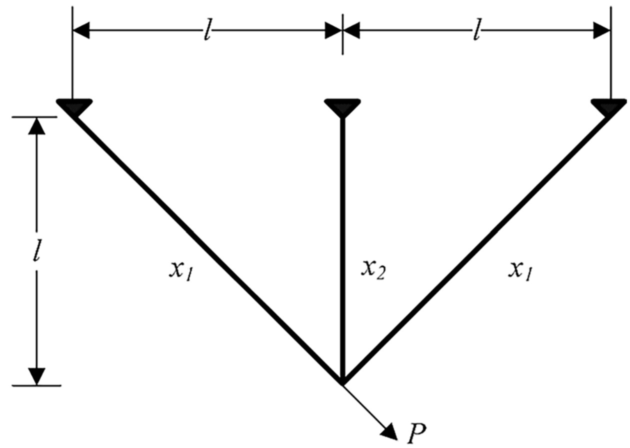


Fig. 8 Three-bar truss design

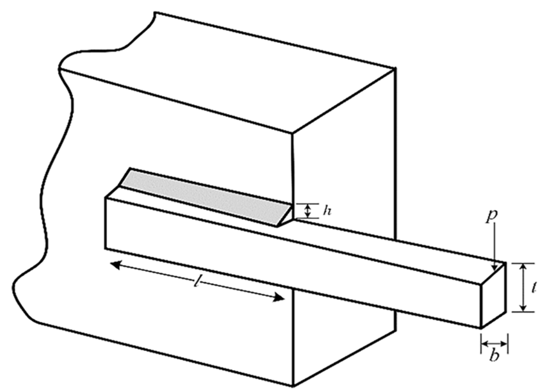


Fig. 9 Welded beam design

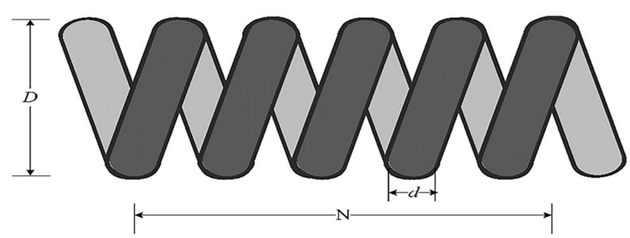
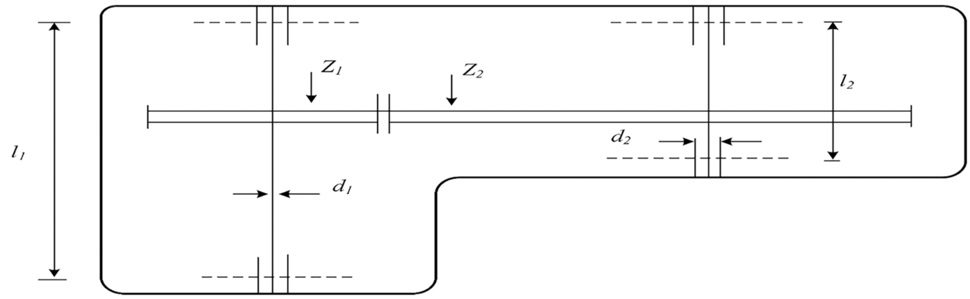


Fig. 10 Tension/compression spring design

Fig. 11 Speed reducer design



Consider $\vec{x} = [x_1 x_2 x_3 x_4] = [T_s T_h R L]$

$$\text{Minimize } f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

$$\text{Subject to } g_1(\vec{x}) = -x_1 + 0.0193x_3 \le 0,\ g_2(\vec{x}) = -x_2 + 0.00954x_3 \le 0,$$
$$g_3(\vec{x}) = -\pi x_3^2 x_4 - \tfrac{4}{3}\pi x_3^3 + 1,296,000 \le 0,\ g_4(\vec{x}) = x_4 - 240 \le 0,$$
$$\text{where } 0 \le x_i \le 100 \text{ for } i = 1,2 \text{ and } 10 \le x_i \le 200 \text{ for } i = 3,4 \tag{24}$$

C.2. Three-Bar Truss Problem

This issue's purpose is to manufacture a truss with the least amount of weight while still adhering to three limitations. Regarding Fig. C.2, two design variables, x_1 and x_2, should be chosen while taking into account limits on stress, deflection, and buckling. Equation (25) is the mathematical representation of this problem.

Consider $\vec{x} = [x_1 x_2] = [A_1 A_2]$

$$\text{Minimize } f(\vec{x}) = \left(2\sqrt{2}x_1 + x_2\right) \times l$$

$$\text{Subject to } g_1(\vec{x}) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}\mathrm{P} - \sigma \le 0,\ g_2(\vec{x}) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}\mathrm{P} - \sigma \le 0,$$

$$g_3(\vec{x}) = \frac{1}{\sqrt{2}x_2 + x_1}\mathrm{P} - \sigma \le 0$$

$$\text{where } 0 \le x_1, x_2 \le 1,\ l = 100\ \text{cm}, P = 2\ \text{kN/cm}^2,\ \sigma = 2\ \text{kN/cm}^2 \tag{25}$$

C.3. Welded Beam Problem

Determining the minimum cost to fabricate a welded beam is the subject of this design problem. It has four design factors that need to be optimized as shown in Fig. C.10 and four restrictions that should be considered. Equation (26) is the mathematical representation of this problem.

Consider $\vec{x} = [x_1 x_2 x_3 x_4] = [hltb]$

$$\text{Minimize } f(\vec{x}) = 1.10471x_1^2x_2 + 0.04811x_3x_4 \times (14.0 + x_2)$$

$$\text{Subject to } g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \le 0, g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \le 0, g_3(\vec{x}) = \delta(\vec{x}) - \delta_{max} \le 0,$$

$$g_4(\vec{x}) = x_1 - x_4 \le 0, g_5(\vec{x}) = \mathrm{P} - P_c(\vec{x}) \le 0, g_6(\vec{x}) = 0.125 - x_1 \le 0$$

$$g_7(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4 \times (14.0 + x_2) - 0.5 \le 0$$

$$\text{where } 0.1 \le x_i \le 2 \text{ for } i = 1,2 \text{ and } 0.1 \le x_i \le 10 \text{ for } i = 3,4 \tag{26}$$

C.4. Tension/compression Spring Design Problem

The major goal of this design problem is to reduce the weight of the tension/compression spring. This problem has three design factors, as shown in Fig. C.11. Equation (27) is the mathematical representation of this problem.

Consider $\vec{x} = [x_1 x_2 x_3] = [dDN]$

$$\text{Minimize } f(\vec{x}) = (x_3 + 2)x_2x_1^2$$

$$\text{Subject to } g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{71785x_1^2} \le 0, g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} \le 0$$

$$g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \le 0, g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \le 0$$

$$\text{where } 0.05 \le x_1 \le 2.00, 0.25 \le x_2 \le 1.30, 2.00 \le x_3 \le 15.0 \tag{27}$$

C.5. Speed Reducer Design Problem

Taking into consideration the bending stress of the gear teeth, the surface stress, the transverse deflections, and the stresses in the shafts, the goal of this restricted optimization issue is to minimize the weight of the speed reducer. This problem has seven variables, as shown in Fig. C.3. The mathematical representation of this problem shown in Eq. (28).

Consider

$$\begin{aligned} \text{Minimize } f(\vec{x}) &= 0.7854x_1x_2^2\left(3.3333x_3^2 + 14.9334x_3 - 43.0934\right) \\ &- 1.508x_1\left(x_6^2 + x_7^2\right) + 7.4777\left(x_6^3 + x_7^3\right) \\ &+ 0.7854\left(x_4x_6^2 + x_5x_7^2\right) \end{aligned}$$

$$\text{Subject to } g_1(\vec{x}) = \frac{27}{x_1x_2^2x_3} - 1 \le 0, \quad g_2(\vec{x}) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \le 0.,$$

$$g_3(\vec{x}) = \frac{1.93x_4^3}{x_2x_6^4x_3} - 1 \le 0$$

$$g_4(\vec{x}) = \frac{1.93x_5^3}{x_2x_7^4x_3} - 1 \le 0, g_5(\vec{x}) = \frac{\left[\left(745(x_4/x_2x_3)\right)^2 + 16.9\times10^6\right]^{1/2}}{110x_6^3} - 1 \le 0,$$

$$g_6(\vec{x}) = \frac{\left[\left(745\left(x_5/x_2x_3\right)\right)^2 + 157.5\times10^6\right]^{1/2}}{85x_7^3} - 1 \le 0$$

$$g_7(\vec{x}) = \frac{x_2x_3}{40} - 1 \le 0, g_8(\vec{x}) = \frac{5x_2}{x_1} - 1 \le 0, g_9(\vec{x}) = \frac{x_1}{12x_2} - 1 \le 0$$

$$g_{10}(\vec{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \le 0, g_{11}(\vec{x}) = \frac{1.1x_7 + 1.9}{x_5} - 1 \le 0,$$

$$\text{where } 2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8, 17 \le x_3 \le 28, 7.3 \le x_4 \le 8.3,$$

$$7.3 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9, 5.0 \le x_7 \le 5.5 \tag{28}$$

Data Availability The datasets used during the current study are available from the corresponding author upon reasonable request.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Talbi, E. G. (2009). *Metaheuristics: from design to implementation* (Vol. 74). Wiley.
2. Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences, 237*, 82–117.
3. Sörensen, K., & Glover, F. (2013). Metaheuristics. *Encyclopedia of Operations Research and Management Science, 62*, 960–970.
4. Bandaru, S., & Deb, K. (2016). Metaheuristic techniques. In *Decision sciences* (pp. 709–766). CRC Press.
5. Gupta, S., Deep, K., & Mirjalili, S. (2020). An efficient equilibrium optimizer with mutation strategy for numerical optimization. *Applied Soft Computing, 96*, 106542.
6. Mohapatra, P., Das, K. N., & Roy, S. (2017). A modified competitive swarm optimizer for large scale optimization problems. *Applied Soft Computing, 59*, 340–362.
7. Kovačević, M., Madić, M., & Radovanović, M. (2013). Software prototype for validation of machining optimization solutions obtained with meta-heuristic algorithms. *Expert Systems with Applications, 40*(17), 6985–6996.
8. Sun, Y., Yang, T., & Liu, Z. (2019). A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems. *Applied Soft Computing, 85*, 105744.
9. Panda, S., & Padhy, N. P. (2008). Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. *Applied Soft Computing, 8*(4), 1418–1427.
10. Panwar, K., & Deep, K. (2021). Discrete Grey Wolf Optimizer for symmetric travelling salesman problem. *Applied Soft Computing, 105*, 107298.
11. Nadimi-Shahraki, M. H., Taghian, S., & Mirjalili, S. (2021). An improved grey wolf optimizer for solving engineering problems. *Expert Systems with Applications, 166*, 113917.
12. Talbi, E.-G. (2021). Machine learning into metaheuristics: A survey and taxonomy. *ACM Computing Surveys (CSUR), 54*(6), 1–32.
13. Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper optimisation algorithm: Theory and application. *Advances in Engineering Software, 105*, 30–47.
14. Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. *Editorial Commentary*, (3), 95–99.
15. Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*(4), 341–359.
16. Rechenberg, I. (1973). Evolution strategy: Optimization of technical systems by means of biological evolution. *Fromman-Holzboog Stuttgart, 104*, 15–16.
17. Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan.
18. Yang, X.-S. (2010). A new metaheuristic bat-inspired algorithm. In J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, & N. Krasnogor (Eds.), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* (pp. 65–74). Berlin, Heidelberg: Springer.
19. Rajabioun, R. (2011). Cuckoo optimization algorithm. *Applied Soft Computing, 11*(8), 5508–5518.
20. Gandomi, A. H., & Alavi, A. H. (2012). Krill herd: A new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation, 17*(12), 4831–4845.
21. Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software, 69*, 46–61.
22. Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems, 89*, 228–249.
23. Arora, S., & Singh, S. (2019). Butterfly optimization algorithm: A novel approach for global optimization. *Soft Computing, 23*(3), 715–734.
24. Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp Swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in*

Engineering Software, 114, 163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002
25. Hashim, F. A., Houssein, E. H., Hussain, K., Mabrouk, M. S., & Al-Atabany, W. (2022). Honey badger algorithm: New metaheuristic algorithm for solving optimization problems. *Mathematics and Computers in Simulation, 192*, 84–110.
26. Houssein, E. H., Oliva, D., Samee, N. A., Mahmoud, N. F., & Emam, M. M. (2023). Liver cancer algorithm: A novel bio-inspired optimizer. *Computers in Biology and Medicine, 165*, 107389.
27. Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software, 95*, 51–67.
28. Haghnegahdar, L., & Wang, Y. (2020). A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection. *Neural Computing and Applications, 32*, 9427–9441.
29. Priyanga, P., Pattankar, V. V., & Sridevi, S. (2021). A hybrid recurrent neural network-logistic chaos-based whale optimization framework for heart disease prediction with electronic health records. *Computational Intelligence, 37*(1), 315–343.
30. Gul, F., Mir, I., Rahiman, W., & Islam, T. U. (2021). Novel implementation of multi-robot space exploration utilizing coordinated multi-robot exploration and frequency modified whale optimization algorithm. *IEEE Access, 9*, 22774–22787.
31. Miao, Y., Zhao, M., Makis, V., & Lin, J. (2019). Optimal swarm decomposition with whale optimization algorithm for weak feature extraction from multicomponent modulation signal. *Mechanical Systems and Signal Processing, 122*, 673–691.
32. Mohammed, H., & Rashid, T. (2020). A novel hybrid GWO with WOA for global numerical optimization and solving pressure vessel design. *Neural Computing and Applications, 32*(18), 14701–14718.
33. Chakraborty, S., Saha, A. K., Sharma, S., Chakraborty, R., & Debnath, S. (2023). A hybrid whale optimization algorithm for global optimization. *Journal of Ambient Intelligence and Humanized Computing, 14*(1), 431–467.
34. Zhang, X., & Wen, S. (2021). Hybrid whale optimization algorithm with gathering strategies for high-dimensional problems. *Expert Systems with Applications, 179*, 115032.
35. Nadimi-Shahraki, M. H., Taghian, S., Mirjalili, S., Abualigah, L., Abd Elaziz, M., & Oliva, D. (2021). EWOA-OPF: Effective whale optimization algorithm to solve optimal power flow problem. *Electronics, 10*(23), 2975.
36. Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation, 1*(1), 67–82.
37. Nadimi-Shahraki, M. H., Taghian, S., Mirjalili, S., & Faris, H. (2020). MTDE: An effective multi-trial vector-based differential evolution algorithm and its applications for engineering design problems. *Applied Soft Computing, 97*, 106761.
38. Awad, N., Ali, M., Liang, J., Qu, B., & Suganthan, P. (2016). Problem definitions and evaluation criteria for the cec 2017 special sessionand competition on single objective real-parameter numerical optimization. In *Nanyang Technological University, Jordan University of Science and Technology and Zhengzhou University, Singapore and Zhenzhou, China, Tech. Rep*, 201611.
39. Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems, 97*, 849–872.
40. Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M., & Gandomi, A. H. (2021). The arithmetic optimization algorithm. *Computer Methods in Applied Mechanics and Engineering, 376*, 113609.
41. Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation, 1*(1), 3–18.
42. Mohamed, A. W., Hadi, A. A., Fattouh, A. M., & Jambi, K. M. (2017). LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems. In *2017 IEEE Congress on evolutionary computation (CEC)*, Donostia, Spain.
43. Awad, N. H., Ali, M. Z., & Suganthan, P. N. (2017). Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems. In *2017 IEEE congress on evolutionary computation (CEC)*, Donostia, Spain.
44. Hashim, F. A., & Hussien, A. G. (2022). Snake optimizer: A novel meta-heuristic optimization algorithm. *Knowledge-Based Systems, 242*, 108320.
45. Dehghani, M., Montazeri, Z., Trojovská, E., & Trojovský, P. (2023). Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems. *Knowledge-Based Systems, 259*, 110011.
46. Nadimi-Shahraki, M. H., Zamani, H., & Mirjalili, S. (2022). Enhanced whale optimization algorithm for medical feature selection: A COVID-19 case study. *Computers in Biology and Medicine, 148*, 105858.
47. Rajesh, C., & Kumar, S. (2022). An evolutionary block based network for medical image denoising using Differential Evolution. *Applied Soft Computing, 121*, 108776.
48. Nadimi-Shahraki, M. H., Taghian, S., Zamani, H., Mirjalili, S., & Elaziz, M. A. (2023). MMKE: Multi-trial vector-based monkey king evolution algorithm and its applications for engineering optimization problems. *PLoS One, 18*(1), e0280006.
49. Abdelkader, E. M., Moselhi, O., Marzouk, M., & Zayed, T. (2022). An exponential chaotic differential evolution algorithm for optimizing bridge maintenance plans. *Automation in Construction, 134*, 104107.
50. Zou, D., & Gong, D. (2022). Differential evolution based on migrating variables for the combined heat and power dynamic economic dispatch. *Energy, 238*, 121664.
51. Zhang, X., Peng, H., Zhang, J., & Wang, Y. (2023). A cost-sensitive attention temporal convolutional network based on adaptive top-k differential evolution for imbalanced time-series classification. *Expert Systems with Applications, 213*, 119073.
52. Wang, G.-G., Deb, S., & Cui, Z. (2019). Monarch butterfly optimization. *Neural Computing and Applications, 31*(7), 1995–2014.
53. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. *Expert Systems with Applications, 174*, 114685.
54. MiarNaeimi, F., Azizyan, G., & Rashki, M. (2021). Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems. *Knowledge-Based Systems, 213*, 106711.
55. Jiang, Y., Wu, Q., Zhu, S., & Zhang, L. (2022). Orca predation algorithm: A novel bio-inspired algorithm for global optimization problems. *Expert Systems with Applications, 188*, 116026.
56. Braik, M., Hammouri, A., Atwan, J., Al-Betar, M. A., & Awadallah, M. A. (2022). White Shark Optimizer: A novel bio-inspired meta-heuristic algorithm for global optimization problems. *Knowledge-Based Systems, 243*, 108457.
57. Zhao, W., Wang, L., & Mirjalili, S. (2022). Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Computer Methods in Applied Mechanics and Engineering, 388*, 114194.
58. Taghian, S., & Nadimi-Shahraki, M. H. (2019). Binary sine cosine algorithms for feature selection from medical data. arXiv preprint arXiv:1911.07805.

59. Erol, O. K., & Eksin, I. (2006). A new optimization method: Big bang–big crunch. *Advances in Engineering Software, 37*(2), 106–111.
60. Kaveh, A., & Talatahari, S. (2010). A novel heuristic optimization method: Charged system search. *Acta Mechanica, 213*(3), 267–289.
61. Kaveh, A., & Khayatazad, M. (2012). A new meta-heuristic method: Ray optimization. *Computers & Structures, 112*, 283–294.
62. Kaveh, A., & Mahdavi, V. R. (2014). Colliding bodies optimization: A novel meta-heuristic method. *Computers & Structures, 139*, 18–27.
63. Zhao, W., Wang, L., & Zhang, Z. (2019). Atom search optimization and its application to solve a hydrogeologic parameter estimation problem. *Knowledge-Based Systems, 163*, 283–304.
64. Wei, Z., Huang, C., Wang, X., Han, T., & Li, Y. (2019). Nuclear reaction optimization: A novel and powerful physics-based algorithm for global optimization. *IEEE Access, 7*, 66084–66109.
65. Kaveh, A., Akbari, H., & Hosseini, S. M. (2020). Plasma generation optimization: A new physically-based metaheuristic algorithm for solving constrained optimization problems. *Engineering Computations, 38*(4), 1554–1606.
66. Rao, R. V., Savsani, V. J., & Vakharia, D. (2011). Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design, 43*(3), 303–315.
67. Moosavi, S. H. S., & Bardsiri, V. K. (2019). Poor and rich optimization algorithm: A new human-based and multi populations algorithm. *Engineering Applications of Artificial Intelligence, 86*, 165–181.
68. Dai, C., Zhu, Y., & Chen, W. (2006). Seeker optimization algorithm. In *International conference on computational and information science*, Guangzhou, China.
69. Wang, C., Zhang, X., Niu, Y., Gao, S., Jiang, J., Zhang, Z., & Dong, H. (2022). Dual-population social group optimization algorithm based on human social group behavior law. *IEEE Transactions on Computational Social Systems, 10*(1), 166–177.
70. Panwar, D., Saini, G., & Agarwal, P. (2022). Human eye vision algorithm (HEVA): A novel approach for the optimization of combinatorial problems. In *Artificial Intelligence in Healthcare* (pp. 61–71). Springer.
71. Houssein, E. H., Helmy, B.E.-D., Elngar, A. A., Abdelminaam, D. S., & Shaban, H. (2021). An improved tunicate swarm algorithm for global optimization and image segmentation. *IEEE Access, 9*, 56066–56092.
72. Houssein, E. H., Helmy, B.E.-D., Oliva, D., Jangir, P., Premkumar, M., Elngar, A. A., & Shaban, H. (2022). An efficient multi-thresholding based COVID-19 CT images segmentation approach using an improved equilibrium optimizer. *Biomedical Signal Processing and Control, 73*, 103401.
73. Yu, X., & Wu, X. (2022). Ensemble grey wolf Optimizer and its application for image segmentation. *Expert Systems with Applications, 209*, 118267.
74. Casas-Ordaz, A., Oliva, D., Navarro, M. A., Ramos-Michel, A., & Pérez-Cisneros, M. (2023). An improved opposition-based Runge Kutta optimizer for multilevel image thresholding. *The Journal of Supercomputing*, 1–108.
75. Nadimi-Shahraki, M. H., Banaie-Dezfouli, M., Zamani, H., Taghian, S., & Mirjalili, S. (2021). B-MFO: A binary moth-flame optimization for feature selection from medical datasets. *Computers, 10*(11), 136.
76. Wang, J., Lin, D., Zhang, Y., & Huang, S. (2022). An adaptively balanced grey wolf optimization algorithm for feature selection on high-dimensional classification. *Engineering Applications of Artificial Intelligence, 114*, 105088.
77. Huang, Y., Li, F., Bao, G., Xiao, Q., & Wang, H. (2022). Modeling the effects of biodiesel chemical composition on iodine value using novel machine learning algorithm. *Fuel, 316*, 123348.
78. Kalita, D. J., Singh, V. P., & Kumar, V. (2022). Two-way threshold-based intelligent water drops feature selection algorithm for accurate detection of breast cancer. *Soft Computing, 26*(5), 2277–2305.
79. Chatterjee, S., Saha, D., Sen, S., Oliva, D., & Sarkar, R. (2023). Moth-flame optimization based deep feature selection for facial expression recognition using thermal images. *Multimedia Tools and Applications*, 1–24.
80. Fang, L., & Liang, X. (2023). A novel method based on nonlinear binary grasshopper whale optimization algorithm for feature selection. *Journal of Bionic Engineering, 20*(1), 237–252.
81. Ramadan, A.-E., Kamel, S., Khurshaid, T., Oh, S.-R., & Rhee, S.-B. (2021). Parameter extraction of three diode solar photovoltaic model using improved Grey Wolf optimizer. *Sustainability, 13*(12), 6963.
82. Xie, Q., Guo, Z., Liu, D., Chen, Z., Shen, Z., & Wang, X. (2021). Optimization of heliostat field distribution based on improved Gray Wolf optimization algorithm. *Renewable Energy, 176*, 447–458.
83. Yesilbudak, M. (2021). Parameter extraction of photovoltaic cells and modules using grey wolf optimizer with dimension learning-based hunting search strategy. *Energies, 14*(18), 5735.
84. Devarapalli, R., Rao, B. V., & Al-Durra, A. (2022). Optimal parameter assessment of solar photovoltaic module equivalent circuit using a novel enhanced hybrid GWO-SCA algorithm. *Energy Reports, 8*, 12282–12301.
85. Nadimi-Shahraki, M. H., Taghian, S., Mirjalili, S., Zamani, H., & Bahreininejad, A. (2022). GGWO: Gaze cues learning-based grey wolf optimizer and its applications for solving engineering problems. *Journal of Computational Science, 61*, 101636.
86. Ma, C., Huang, H., Fan, Q., Wei, J., Du, Y., & Gao, W. (2022). Grey wolf optimizer based on aquila exploration method. *Expert Systems with Applications, 205*, 117629.
87. Pan, J.-S., Zhang, L.-G., Wang, R.-B., Snášel, V., & Chu, S.-C. (2022). Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems. *Mathematics and Computers in Simulation, 202*, 343–373.
88. Yuan, Y., Shen, Q., Wang, S., Ren, J., Yang, D., Yang, Q., Mu, X. (2023). Coronavirus mask protection algorithm: A new bio-inspired optimization algorithm and its applications. *Journal of Bionic Engineering*, 1–19.
89. Mesquita, R., & Gaspar, P. D. (2021). A novel path planning optimization algorithm based on particle swarm optimization for UAVs for bird monitoring and repelling. *Processes, 10*(1), 62.
90. Javaheri, D., Gorgin, S., Lee, J.-A., & Masdari, M. (2022). An improved discrete harris hawk optimization algorithm for efficient workflow scheduling in multi-fog computing. *Sustainable Computing: Informatics and Systems, 36*, 100787.
91. Wu, J., Zhang, P.-W., Wang, Y., & Shi, J. J. (2022). Integrated aviation model and metaheuristic algorithm for hub-and-spoke network design and airline fleet planning. *Transportation Research Part E: Logistics and Transportation Review, 164*, 102755.
92. Sa, A., Yv, R. R., & Sadiq, A. S. (2022). Traffic flow forecasting using natural selection based hybrid Bald Eagle Search—Grey Wolf optimization algorithm. *PLoS One, 17*(9), e0275104.
93. Kaur, N., Kaur, L., & Cheema, S. S. (2021). An enhanced version of Harris Hawks optimization by dimension learning-based hunting for breast cancer detection. *Scientific Reports, 11*(1), 1–26.
94. Mohakud, R., & Dash, R. (2022). Skin cancer image segmentation utilizing a novel EN-GWO based hyper-parameter optimized

FCEDN. *Journal of King Saud University-Computer and Information Sciences*.
95. Nadimi-Shahraki, M. H., Taghian, S., Mirjalili, S., & Abualigah, L. (2022). Binary aquila optimizer for selecting effective features from medical data: A COVID-19 case study. *Mathematics, 10*(11), 1929.
96. Duan, Y., Liu, C., Li, S., Guo, X., & Yang, C. (2021). Manta ray foraging and Gaussian mutation-based elephant herding optimization for global optimization. *Engineering with Computers*, 1–41.
97. Jia, H., Sun, K., Zhang, W., & Leng, X. (2022). An enhanced chimp optimization algorithm for continuous optimization domains. *Complex & Intelligent Systems, 8*(1), 65–82.
98. Nadimi-Shahraki, M. H., Taghian, S., Mirjalili, S., Ewees, A. A., Abualigah, L., & Abd Elaziz, M. (2021). Mtv-mfo: Multi-trial vector-based moth-flame optimization algorithm. *Symmetry, 13*(12), 2388.
99. Zhong, C., & Li, G. (2022). Comprehensive learning Harris hawks-equilibrium optimization with terminal replacement mechanism for constrained optimization problems. *Expert Systems with Applications, 192*, 116432.
100. Singh, H., Singh, B., & Kaur, M. (2022). An improved elephant herding optimization for global optimization problems. *Engineering with Computers, 38*(4), 3489–3521.
101. Wang, Z., Ding, H., Yang, Z., Li, B., Guan, Z., & Bao, L. (2022). Rank-driven salp swarm algorithm with orthogonal opposition-based learning for global optimization. *Applied Intelligence, 52*(7), 7922–7964.
102. Ma, J., Hao, Z., & Sun, W. (2022). Enhancing sparrow search algorithm via multi-strategies for continuous optimization problems. *Information Processing & Management, 59*(2), 102854.
103. Nadimi-Shahraki, M. H. (2023). An effective hybridization of quantum-based avian navigation and bonobo optimizers to solve numerical and mechanical engineering problems. *Journal of Bionic Engineering, 20*(3), 1361–1385.
104. Hosseini, E., Sadiq, A. S., Ghafoor, K. Z., Rawat, D. B., Saif, M., & Yang, X. (2021). Volcano eruption algorithm for solving optimization problems. *Neural Computing and Applications, 33*, 2321–2337.
105. Meng, A., Zeng, C., Wang, P., Chen, D., Zhou, T., Zheng, X., & Yin, H. (2021). A high-performance crisscross search based grey wolf optimizer for solving optimal power flow problem. *Energy, 225*, 120211.
106. Shaheen, M. A., Hasanien, H. M., & Al-Durra, A. (2021). Solving of optimal power flow problem including renewable energy resources using HEAP optimization algorithm. *IEEE Access, 9*, 35846–35863.
107. Jiang, Y., Wu, Q., Zhang, G., Zhu, S., & Xing, W. (2021). A diversified group teaching optimization algorithm with segment-based fitness strategy for unmanned aerial vehicle route planning. *Expert Systems with Applications, 185*, 115690.
108. Liu, P., Hendalianpour, A., Feylizadeh, M., & Pedrycz, W. (2022). Mathematical modeling of vehicle routing problem in omni-channel retailing. *Applied Soft Computing, 131*, 109791.
109. Nadimi-Shahraki, M. H., Moeini, E., Taghian, S., & Mirjalili, S. (2021). DMFO-CD: A discrete moth-flame optimization algorithm for community detection. *Algorithms, 14*(11), 314.
110. Koc, I. (2022). A fast community detection algorithm based on coot bird metaheuristic optimizer in social networks. *Engineering Applications of Artificial Intelligence, 114*, 105202.
111. Nadimi-Shahraki, M. H., Moeini, E., Taghian, S., & Mirjalili, S. (2023). Discrete improved grey wolf optimizer for community detection. *Journal of Bionic Engineering*, 1–28.
112. Hashemi, M., Javaheri, D., Sabbagh, P., Arandian, B., & Abnoosian, K. (2021). A multi-objective method for virtual machines allocation in cloud data centres using an improved grey wolf optimization algorithm. *IET Communications, 15*(18), 2342–2353.
113. Gao, Y., Yang, B., Wang, S., Zhang, Z., & Tang, X. (2022). Bi-objective service composition and optimal selection for cloud manufacturing with QoS and robustness criteria. *Applied Soft Computing, 128*, 109530.
114. Sing, R., Bhoi, S. K., Panigrahi, N., Sahoo, K. S., Jhanjhi, N., & AlZain, M. A. (2022). A whale optimization algorithm based resource allocation scheme for cloud-fog based IoT applications. *Electronics, 11*(19), 3207.
115. Cheng, M.-Y., & Prayogo, D. (2014). Symbiotic organisms search: A new metaheuristic optimization algorithm. *Computers & Structures, 139*, 98–112.
116. Fan, Q., Chen, Z., Zhang, W., & Fang, X. (2020). ESSAWOA: Enhanced whale optimization algorithm integrated with salp swarm algorithm for global optimization. *Engineering with Computers*, 1–18.
117. Singh, A. (2019). Laplacian whale optimization algorithm. *International Journal of System Assurance Engineering and Management, 10*(4), 713–730.
118. Yan, Z., Zhang, J., Zeng, J., & Tang, J. (2021). Nature-inspired approach: An enhanced whale optimization algorithm for global optimization. *Mathematics and Computers in Simulation, 185*, 17–46.
119. Saafan, M. M., & El-Gendy, E. M. (2021). IWOSSA: An improved whale optimization salp swarm algorithm for solving optimization problems. *Expert Systems with Applications, 176*, 114901.
120. Kalananda, V. K. R. A., & Komanapalli, V. L. N. (2021). A combinatorial social group whale optimization algorithm for numerical and engineering optimization problems. *Applied Soft Computing, 99*, 106903.
121. Satapathy, S., & Naik, A. (2016). Social group optimization (SGO): A new population evolutionary optimization technique. *Complex & Intelligent Systems, 2*(3), 173–203.
122. Chen, H., Li, W., & Yang, X. (2020). A whale optimization algorithm with chaos mechanism based on quasi-opposition for global optimization problems. *Expert Systems with Applications, 158*, 113612.
123. Fan, Q., Chen, Z., Li, Z., Xia, Z., Yu, J., & Wang, D. (2021). A new improved whale optimization algorithm with joint search mechanisms for high-dimensional global optimization problems. *Engineering with Computers, 37*(3), 1851–1878.
124. Liang, X., & Zhang, Z. (2022). A whale optimization algorithm with convergence and exploitability enhancement and its application. *Mathematical Problems in Engineering*, 2022.
125. Sun, G., Shang, Y., & Zhang, R. (2022). An efficient and robust improved whale optimization algorithm for large scale global optimization problems. *Electronics, 11*(9), 1475.
126. Yang, W., Xia, K., Fan, S., Wang, L., Li, T., Zhang, J., & Feng, Y. (2022). A multi-strategy Whale optimization algorithm and its application. *Engineering Applications of Artificial Intelligence, 108*, 104558.
127. Kannan, B., & Kramer, S. N. (1994). An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. *Journal of Mechanical Design, 116*(2), 405–411.
128. Nowacki, H. (1974). Optimization in pre-contract ship design. In: Fujita, Y., Lind, K., Williams, T.J. (eds) *Computer Applications in the Automation of Shipyard Operation and Ship Design*, vol. 2 (pp. 327–338).

129. Coello, C. A. C. (2000). Use of a self-adaptive penalty approach for engineering optimization problems. *Computers in Industry, 41*(2), 113–127.
130. Arora, J. S. (2004). *Introduction to optimum design*. Elsevier.
131. Golinski, J. (1973). An adaptive optimization system applied to machine synthesis. *Mechanism and Machine Theory, 8*(4), 419–436.