Neural Computing and Applications
https://doi.org/10.1007/s00521-022-07530-9

ORIGINAL ARTICLE

Prairie Dog Optimization Algorithm

Absalom E. Ezugwu[1] · Jeffrey O. Agushaka[1,2] · Laith Abualigah[3,4,5] · Seyedali Mirjalili[6,7] · Amir H. Gandomi[8]

Received: 12 February 2022 / Accepted: 9 June 2022

Abstract

This study proposes a new nature-inspired metaheuristic that mimics the behaviour of the prairie dogs in their natural habitat called the prairie dog optimization (PDO). The proposed algorithm uses four prairie dog activities to achieve the two common optimization phases, exploration and exploitation. The prairie dogs' foraging and burrow build activities are used to provide exploratory behaviour for PDO. The prairie dogs build their burrows around an abundant food source. As the food source gets depleted, they search for a new food source and build new burrows around it, exploring the whole colony or problem space to discover new food sources or solutions. The specific response of the prairie dogs to two unique communication or alert sound is used to accomplish exploitation. The prairie dogs have signals or sounds for different scenarios ranging from predator threats to food availability. Their communication skills play a significant role in satisfying the prairie dogs' nutritional needs and anti-predation abilities. These two specific behaviours result in the prairie dogs converging to a specific location or a promising location in the case of PDO implementation, where further search (exploitation) is carried out to find better or near-optimal solutions. The performance of PDO in carrying out optimization is tested on a set of twenty-two classical benchmark functions and ten CEC 2020 test functions. The experimental results demonstrate that PDO benefits from a good balance of exploration and exploitation. Compared with the results of other well-known population-based metaheuristic algorithms available in the literature, the PDO shows stronger performance and higher capabilities than the other algorithms. Furthermore, twelve benchmark engineering design problems are used to test the performance of PDO, and the results indicate that the proposed PDO is effective in estimating optimal solutions for real-world optimization problems with unknown global optima. The PDO algorithm source codes is publicly available at https://www.mathworks.com/matlabcentral/fileexchange/110980-prairie-dog-optimization-algorithm.

Keywords Nature-inspired · Prairie dogs · Optimization algorithm · Anti-predation · Burrow-building · Optimization · Swarm intelligence

1 Introduction

The goal of optimization is finding the best solution to a problem given some constraints and objectives. Optimization problems are characterized by unknown search space, discrete or continuous search space, non-derivative objective functions, high dimensions, non-convexity, and many more, which cannot be solved within minimum time or accuracy complexity by traditional classical methods [1]. Optimization techniques can be broadly classified into deterministic and non-deterministic (stochastic) methods. Some deterministic methods are typically gradient-based and are further divided into linear and nonlinear approaches [2]. Though they help solve linear and nonlinear, differentiable optimization problems, they suffer from being trapped in the local optima [3, 4]. Modifying the population's initial position and hybridizing different algorithms have been proposed as solutions to being trapped in the local optima [5, 6]. The non-deterministic (stochastic) methods utilize randomly generated variables to search the problem space for near-optimal solutions. They are usually simple, independent, flexible, and gradient-free, which is a great advantage for these methods [7, 8].

So many principles and processes in nature have been used as inspiration for developing metaheuristic algorithms used to solve optimization problems [4, 9]. The past two

Extended author information available on the last page of the article

Published online: 15 July 2022

decades have witnessed a significant rise in the number of proposed nature-inspired optimization algorithms, and some imitate biological behaviour, physical phenomena, and many more [9]. The earliest and one of the renowned population-based metaheuristic algorithms is the genetic algorithm (GA), based on the concept of survival of the fittest specified in the Darwinian theory of evolution [10]. Another renowned swarm-based population-based metaheuristic algorithm is particle swarm optimization (PSO) [11]. The simulated annealing (SA) [12], ant colony optimization (ACO) [13], and artificial bee colony (ABC) [14] are other popular innovative population-based metaheuristic algorithms that have been used for solving the complex optimization problems inspired by the annealing phenomenon of metallurgy, the social behaviour of birds, colonizing behaviour of ants and bees, respectively.

The popularity of metaheuristic algorithms can be attributed to ease of comprehension and implementation, good performance (finding optimal or near-optimal solutions), and the ability to avoid being trapped in local optima. The optimization process usually consists of the exploration and exploitation phases [15]. The exploration phase involves performing a global search of the problem search space to identify promising areas to focus on in searching for optimal solutions. At the same time, the exploitation involves performing a local search of the identified promising areas to find better solutions [16]. Maintaining the right balance between exploration and exploitation is the goal of all metaheuristic algorithms because any imbalance may lead to being trapped in local optima or minima [17]. Metaheuristic algorithms have been successfully applied to solve optimization problems in various fields of science, such as process control [18], feature selection [19], biomedical signal processing [20], image processing [21–23], parallel machines job scheduling [24–26], and engineering design problems [27].

The search process of metaheuristic algorithms is stochastic, so the results obtained are near-optimal or may not be the same as the known global optimum solution. Due to this randomness, the results obtained are always different, so it is recommended to run the algorithm independently several times and take the average result for comparison. Also, the stochastically generated solutions are based on the given constraints of the problem, and the solutions are iteratively improved based on the defined updating steps of the algorithm. The difference between one metaheuristic algorithm and another is in the updating process used to improve the candidate solutions during the optimization process of the algorithm.

Metaheuristic algorithms may be categorized into four groups based on the source of inspiration: swarm-based, evolutionary-based, physical-based, and human activity-based algorithms [28–30]. Biological phenomena like natural evolution inspire evolutionary-based algorithms. The GA is one of the most popular algorithms in this category; others include evolutionary programming [31], differential evolution [32], and evolution strategy [33, 34]. The swarm-based algorithms mimic animals' social behaviour in herds, flocks, or schools. The most popular algorithm in this category is particle swarm optimization (PSO); others include the red fox optimization algorithm [35], salp swarm algorithm [36], ant colony optimization [13], and artificial bee colony [14]. The physical-based algorithms mimic some physical laws in nature. One of the most popular algorithms in this category is the simulated annealing; others include the gravitational search algorithm [37], Archimedes optimization algorithm [38], and thermal exchange optimization [39]. Human activity-based algorithms are inspired by human-invented activities such as teaching–learning-based optimization [40] and the imperialist competitive algorithm [41].

The rate at which new algorithms are proposed may suggest that researchers have abandoned efforts at improving existing ones. For instance, the performance of DE was improved by modifying the two control parameters of DE by an embedded switching mechanism [42]. Also, the slow convergence rate identified in ABC was solved by an improved variant of ABC called IABC (improved ABC) [43]. The same issue with ABC was solved by the authors in [44], where they incorporated a modified global best guiding mechanism that enhanced the exploitation ability of ABC. Many authors have used chaos theory, fuzzy logic, and other mathematical models to improve metaheuristic algorithms' diversity, exploration, and exploitation capability. For example, the low convergence problem of the basic cuckoo search algorithm was resolved by integrating the chaos mechanism [45], and the harmony search algorithm was improved by the fuzzy harmony search algorithm [46].

With the number of proposed metaheuristic algorithms in literature and the existence of some prominent metaheuristic algorithms, it begs to answer the need for a new metaheuristic algorithm. However, emerging challenging optimization problems and the need to provide robust algorithms that will achieve better results continuously have been pushing for an improved version of existing algorithms or new ones entirely, also following the "no free lunch" (NFL) theorem [47], which suggest that no single optimization technique can optimally solve all optimization problems. The implication is that the robustness and effectiveness of an optimization algorithm are limited to solving a certain set of problems but not all other classes of problems. Relying on the NFL theorem, many researchers have proposed new nature-inspired metaheuristic algorithms or improved existing ones with varying levels of success.

Hence, the present study proposes a new nature-inspired algorithm called the prairie dog optimization (PDO) for solving unconstrained numerical optimization problems. The proposed algorithm simulates four prairie dogs' activities to achieve optimization. The prairie dogs' foraging and burrow build activities are used to explore the optimization problem space. The prairie dogs build their burrows around an abundant food source. As the food source gets depleted, they search for a new food source and build new burrows around it, exploring the whole colony or problem space to discover new food sources or solutions. The specific response of the prairie dogs to two unique communication or alert sound is used to accomplish exploitation. The prairie dogs have signals or sounds for different scenarios ranging from predator threats to food availability. Their communication skills play a significant role in satisfying the prairie dogs' nutritional needs and anti-predation abilities. These two specific behaviours result in the prairie dogs converging to a specific location or a promising location in the case of PDO implementation, where further search (exploitation) is carried out to find better or near-optimal solutions. The main contributions of the proposed algorithm are as follows:

i. A novel nature-inspired prairie dog optimization (PDO) algorithm is proposed. The feeding, burrow-building, and unique response to specific alarm behaviours of prairie dogs were studied thoroughly and modelled mathematically to carry out the optimization process.
ii. The proposed PDO algorithm is validated on 22 classical benchmark functions and 10 CEC 2020 test functions.
iii. A detailed comparative study of the PDO is performed with other existing state-of-the-art nature-inspired optimization algorithms. The mean and standard deviation were used for statistical analysis (Freidman's test) and the convergence rate analysis.
iv. The robustness and effectiveness of PDO and other optimizers are investigated for optimizing 12 engineering optimization problems.

The remainder of this paper is organized as follows. Section 2 discusses the inspirational source of the proposed PDO algorithm. Section 3 provides the mathematical formulation and optimization procedures of the PDO algorithm. Section 4 evaluates the efficiency of the PDO in optimizing different benchmark test functions and similarly assesses the algorithm's ability to solve several engineering design problems. Finally, Sect. 5 presents the conclusions and valuable suggestions for future research.

2 Inspiration of PDO

The prairie dogs (genus Cynomys) are herbivorous burrowing rodents primarily found in the Great Plains and south-western desert grasslands of the USA and regions around the plains and plateaus of Canada and Mexico [48]. Prairie dogs are related to the Sciuridae (squirrel) family members like ground squirrels and chipmunks. There are five species of the prairie dogs: black-tailed, Gunnison's, white-tailed, Utah, and Mexican, the black-tailed (most common) are the species found in Badlands National Park [49]. A prairie dog weighs 1–3 pounds and can grow up to 14–17 inches in length. The prairie dogs have developed some physical adaptations to survive in their habitats, such as short, strong arms, and long-nailed toes. These adaptations help them dig burrows and escape predators by running up to 35 mph at short distances to the safety of their burrows [50].

2.1 Habitat and burrowing

The regions where the prairie dogs live have altitudes ranging from 2,000 to 10,000 ft above sea level, temperatures ranging from 38 °C (100 °F) in the summer to as cold as − 37 °C (− 35 °F) in the winter, and are susceptible to environmental threats such as drought, prairie fires, floods, hailstorms, and blizzards [51]. The prairie dogs live in burrows that provide essential protection against environmental threats and help control their body temperature. There are between 10 and 100 burrow entrances per acre of the colony, and Fig. 1a shows how the burrows span around the colony. The burrows (tunnel systems) also have ecological importance. It prevents erosion by channelling rainwater into the water table and reverses soil compaction by changing the soil composition in a region [48].

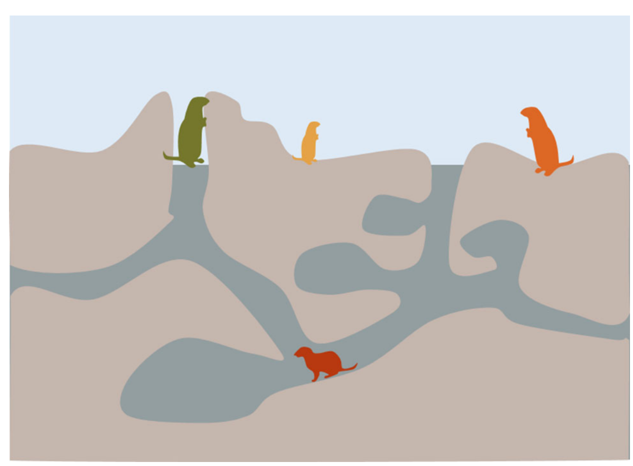


Fig. 1 Prairie dog burrow

The burrows (Fig. 1) have a length of between 5 and 10 m and a depth of between 2 and 3 m, and the burrow entrances (up to six) are usually 10–30 cm in diameter [50]. The burrow entrances are usually simply flat holes in the ground, and sometimes the prairie dogs build mounds of soil around the burrows. The complexity and size of the burrows differ; the dome craters are 20–30 cm high, whereas the rim craters are 1 m high. Both craters serve as an observation post to look out for predators, and they also serve as protection from flooding and aeration of the tunnel [52]. Also, the burrows have different units for different purposes: nursery chambers for their younglings, listening posts, chambers for night and winter, storage rooms, and back doors for different escape routes, as shown in Fig. 1.

2.2 Social organization

Prairie dogs are highly sociable and live together in underground colonies or towns. To adapt to their natural habitat, they build their colonies underground [53]. The prairie dogs are known to live in large colonies (1–1000 acres), and it has been reported of a colony that is 25,000 square miles [48]. No matter the colony's size, the complexity and the functionality of the subunits of the colony are the same. The colony or town may house 15–26 family units or coteries and be divided into wards, with each family unit living in wards [51].

A coterie consists of an adult male, two or three adult females, and their offspring (both male and female) and usually occupies areas up to 1 acre [49]. Members of the same coterie live in the same ward and will kiss or sniff upon each other as a means of identification. They do not perform the kiss or sniff behaviours with prairie dogs from another coterie. The female offspring usually remain in their birth coterie for life, thereby ensuring stability in the groups [51]. On the other hand, the male offspring would typically leave their coterie when they are sexually mature to find another family group. Prairie dogs are highly territorial and mark out well-established borders corresponding to physical barriers such as rocks and trees. The alpha male in a coterie defends its territory, fighting off a contending male from another family.

2.3 Communication and anti-predation

The way of communication of the prairie dogs is so fascinating; at least 12 distinct calls that correspond to different postures and displays have been identified [54]. The research community believes that the prairie dogs have one of the most sophisticated and complex languages ever decoded for an animal [55]. Figure 2 shows a prairie dog calling by standing on its hind legs and making squeaky sounds. The sound made by the prairie dog may sound like a simple squeak or yip to humans; it is a detailed message to a prairie dog's ear [56]. Slobodchikoff et al. [54] postulate that the communication skills of the prairie dogs are so sophisticated that they could describe specific predators. The sounds communicate information about food sources, predators, and many more. For instance, while foraging, the prairie dogs frequently stand on their hind legs or lift their heads to look out for danger. Once a danger is spotted, the prairie dogs begin making nasal yips and dive into a burrow. All coterie members will heed the call, look for the source of danger, and take cover depending on the type of danger [55].

Fig. 2 Prairie dog communication with sound pattern

The anti-predation skills of the prairie dog are pretty unique and sophisticated. Armed with dichromatic colour vision, they can detect predators far away from them and notify other prairie dogs of the danger [56]. The unique high-pitched sound made by the prairie dogs can tell other prairie dogs the specific predator and how fast it is approaching. The prairie dogs also have different reactions to different predators. If the alarm is for a hawk attack, all the prairie dogs nearby will stop whatever they are doing and, standing on their hind legs, will look for where the attack is coming. Those in the attack line of the hawk would dive into their burrows, while others stand in their burrow entrance and watch. Detecting a human, all nearby prairie dogs dive inside their burrows. For coyotes, the coterie members rush to the burrow entrance and watch [55].

3 Prairie dog optimization

This section illustrates the mathematical model formulation and PDO algorithm optimization procedures.

3.1 Assumptions and implementation

Typically, the prairie dog's coterie days are spent eating, keeping an eye out for predators, building new burrows, or maintaining existing ones. The optimization process starts with the different coterie foraging from one food source to another. Prairie dogs are chiefly herbivorous, though they eat some insects. They move from one location to another during different seasons, feeding primarily on grasses, tiny seeds, and some insects, thereby exploring the problem search space. They keep an eye out for predators by lifting their heads or standing on their hind legs as they forage. The coterie searches for places with no burrows but strategically located to build new burrows. The new burrows' location must serve a specific purpose that enhances the overall functioning of the town, ward, or coterie. This further enhances exploration. One unique characteristic of prairie dog behaviour is communication skills, and the specific response to the unique sounds helps achieve exploitation for the proposed algorithm. It is generally believed in the scientific community that prairie dogs have one of the most complex languages ever decoded for animals.

To humans, the prairie dog's "bark" is nothing but a simple squeaky or yippee sound; however, it carries much more meaning to a prairie dog's ear. The prairie dogs have signals or sounds for different scenarios ranging from predator threats to food availability. Their communication skills play a significant role in the anti-predation abilities of the prairie dogs. They can communicate the difference between different predators and their hunting patterns. It is believed that this behavioural adaptation arises from a need to have a different response or survival skills to the different hunting strategies of predators. For instance, if the communication specifies a hawk as the predator, only prairie dogs on the flight path of the hawk go into hiding, while others watch from their burrows. For coyotes, they only observe it from the entrances of their respective burrows, while they all run into their burrows for human predators. The specific response to a specific sound is used to achieve exploitation. The different specific sounds are likened to calls to promising regions, and the different coterie moves towards it by responding appropriately. This continuous cycle of behaviours forms the foundation of the proposed prairie dog optimization (PDO). The following assumptions were made to enable the formulation of the models for the proposed PDO:

i. There are n number of prairie dogs in a coterie and m number of coteries in the colony, and every prairie dog belongs to a coterie.
ii. All prairie dogs are the same and are further categorized into m groups.
iii. The colony, which corresponds to the problem search space, is divided into wards and every coterie lives in a ward.
iv. There are at least ten burrows entrances that grow to 100 per ward because of nest-building activities.
v. Two (2) distinct sounds are implemented, an anti-predation and new food source (new burrow building) call.
vi. The foraging and burrow building activities (exploration), communication, and anti-predation (exploitation) activities involve only members of the same coterie.
vii. Since other coteries in the colony do the same activities simultaneously and the entire colony or problem space has been divided into wards (coteries), the exploration and exploitation activities are repeated m (number of coteries) times.

The prairie dog optimization (PDO) follows the random initialization of the location of the prairie dogs, just like other population-based algorithms. The prairie dog's populations are the search agents, and a vector in d-dimensional space is used to represent the location of the individual prairie dog.

3.2 Initialization

There is n number of prairie dogs (PD) in a coterie, and each PD belongs to m coterie. The PD live and function as a unit or coterie; therefore, the location of ith prairie dog in a particular coterie can be specified by a vector. The matrix given in Eq. 1 represents the location of all coteries (CT) in a colony:

$$CT = \begin{bmatrix} CT_{1,1} & CT_{1,2} & \cdots & CT_{1,d-1} & CT_{1,d} \\ CT_{2,1} & CT_{2,2} & \cdots & CT_{2,d-1} & CT_{2,d} \\ \vdots & \vdots & CT_{i,j} & \vdots & \vdots \\ CT_{m,1} & CT_{m,2} & \cdots & CT_{m,d-1} & CT_{m,d} \end{bmatrix} \tag{1}$$

where $CT_{i,j}$ represents the jth dimension of the ith coterie in a colony. Equation 2 represents the location of all the prairie dogs in a coterie:

$$PD = \begin{bmatrix} PD_{1,1} & PD_{1,2} & \cdots & PD_{1,d-1} & PD_{1,d} \\ PD_{2,1} & PD_{2,2} & \cdots & PD_{2,d-1} & PD_{2,d} \\ \vdots & \vdots & PD_{i,j} & \vdots & \vdots \\ PD_{n,1} & PD_{n,2} & \cdots & PD_{n,d-1} & PD_{n,d} \end{bmatrix} \tag{2}$$

where $PD_{i,j}$ represents the jth dimension of the ith prairie dog in a coterie and $n \leq m$. Each coterie and prairie dog location is allocated using a uniform distribution as shown in Eqs. 3 and 4, respectively.

$$CT_{i,j} = U(01) \times \left(UB_j - LB_j\right) + LB_j \tag{3}$$

$$PD_{i,j} = U(0,1) \times \left(ub_j - lb_j\right) + lb_j \tag{4}$$

where $UB_j andLB_j$ are the upper and lower bounds of the jth dimension of the optimization problem, $ub_j = \frac{UB_j}{m}$ $and\ lb_j = \frac{LB_j}{m}$, and U(0,1) is a random number with a uniform distribution between 0 and 1.

3.3 Fitness function evaluation

The fitness function's value for each prairie dog's location is evaluated by feeding the solution vector into the defined fitness function. The resulting values are stored in the array given in Eq. 5.

$$f(PD) = \begin{bmatrix} f_1\left(\begin{bmatrix} PD_{1,1} & PD_{1,2} & \cdots & PD_{1,d-1} & PD_{1,d} \end{bmatrix}\right) \\ f_2\left(\begin{bmatrix} PD_{2,1} & PD_{2,2} & \cdots & PD_{2,d-1} & PD_{2,d} \end{bmatrix}\right) \\ \vdots \quad \vdots \quad \cdots \quad \vdots \quad \vdots \\ f_n\left(\begin{bmatrix} PD_{n,1} & PD_{n,2} & \cdots & PD_{n,d-1} & PD_{n,d} \end{bmatrix}\right) \end{bmatrix} \tag{5}$$

The value of each prairie dog's fitness function represents the quality of food found at a particular source, the potential for building new burrows, and responding correctly to anti-predation alerts. The array storing the fitness function values is sorted, and the minimum fitness value obtained is declared the best solution so far for the given minimization problem. The next three are considered along with the best value for burrow building which aids their escape from predators.

3.4 Exploration

In this section, the exploratory process of PDO is introduced. The prairie dogs' foraging and burrow building activities are used to explore the optimization problem space. The prairie dogs build their burrows around an abundant food source. As the food source gets depleted, they search for a new food source and build new burrows around it, exploring the whole colony or problem space to discover new food sources or solutions. The burrows are essential for environmental and predator protection. As mentioned earlier, the entire prairie dog population lives in towns or colonies, and each colony comprises family units or coteries with clearly defined boundaries within the colony. The different coteries forage and build burrows within their boundaries as a unit, only interrupted by the presence of a predator.

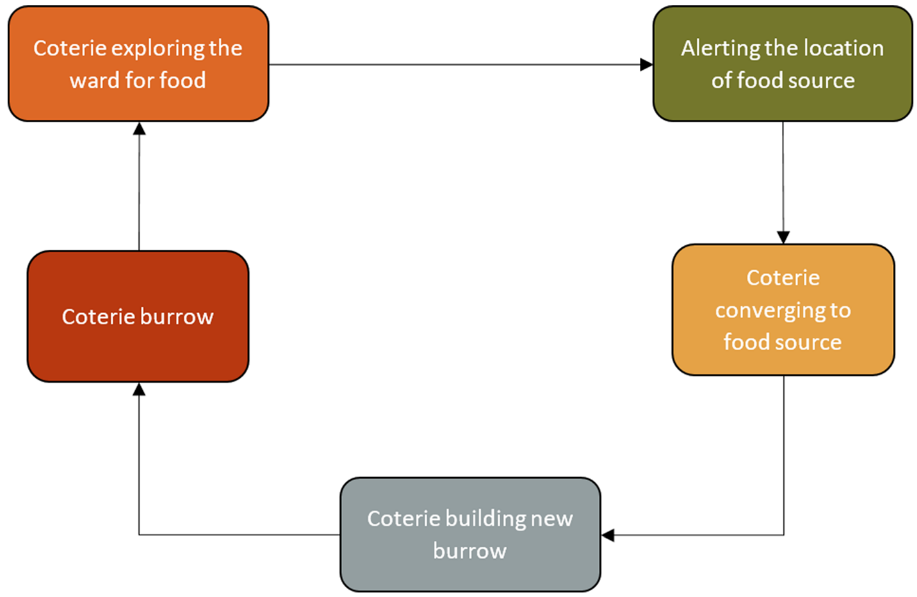
The PDO can navigate between exploration and exploitation based on four conditions. The maximum number of iterations is divided into four; the first two parts are taken by the exploration and the last two parts by the exploitation. The two strategies used for exploration are conditioned on $iter < \frac{Max_{iter}}{4}$ and $\frac{Max_{iter}}{4} \leq iter < \frac{Max_{iter}}{2}$, while the two strategies for exploitation are conditioned on $\frac{Max_{iter}}{2} \leq iter < 3\frac{Max_{iter}}{4}$ and $3\frac{Max_{iter}}{4} \leq iter \leq Max_{iter}$.

As shown in Fig. 3, the first strategy is for coterie members to scour the ward for new food sources in the exploration phase. The Lévy flight motion best captures the movement of the prairie dogs as they search for food sources. This movement does not allow for an intensive search of a particular region because of the characteristic long jumps but ensures a wide range of regions are searched effectively (exploration). Once the food sources are discovered, they make distinctive sounds to communicate the discovery to other members. The quality of the food source is accessed, the best is selected for foraging, and new burrows are built depending on the quality of the food source. The position updating for foraging in the exploration phase of our algorithm is given in Eq. 6.

The second strategy is to assess the quality of the food sources so far and evaluate the digging strength. New burrows are built based on the digging strength designed to diminish as the number of iterations increases. This condition helps to restrict the number of burrows that can be built. The position updating for the burrow building is given in Eq. 7.

$$PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \rho - CPD_{i,j} \times Levy(n)\ \forall\ iter < \frac{Max_{iter}}{4} \tag{6}$$

Fig. 3 Model of the exploration phase



$$PD_{i+1,j+1} = GBest_{i,j} \times rPD \times DS \times Levy(n) \, \forall \, \frac{Max_{iter}}{4} \le iter < \frac{Max_{iter}}{2} \tag{7}$$

where $GBest_{i,j}$ is the global best-obtained solution so far, $eCBest_{i,j}$ evaluates the effect of the current obtained best solution as shown in Eq. 8, ρ is the specialized food source alarm fixed at 0.1 kHz for this experiment, rPD is the position of a random solution, $CPD_{i,j}$ is the randomized cumulative effect of all prairie dogs in the colony and is defined in Eq. 9. DS represents the coterie's digging strength, which depends on the quality of the food source and takes a random value defined by Eq. 10. Lévy(n) is a Lévy distribution, and it is known to encourage better and more efficient exploration of the problem search space [57].

$$eCBest_{i,j} = GBest_{i,j} \times \Delta + \frac{PD_{i,j} \times mean\left(PD_{n,m}\right)}{GBest_{i,j} \times \left(UB_j - LB_j\right) + \Delta} \tag{8}$$

$$CPD_{i,j} = \frac{GBest_{i,j} - rPD_{i,j}}{GBest_{i,j} + \Delta} \tag{9}$$

$$DS = 1.5 \times r \times \left(1 - \frac{iter}{Max_{iter}}\right)^{\left(2\frac{iter}{Max_{iter}}\right)} \tag{10}$$

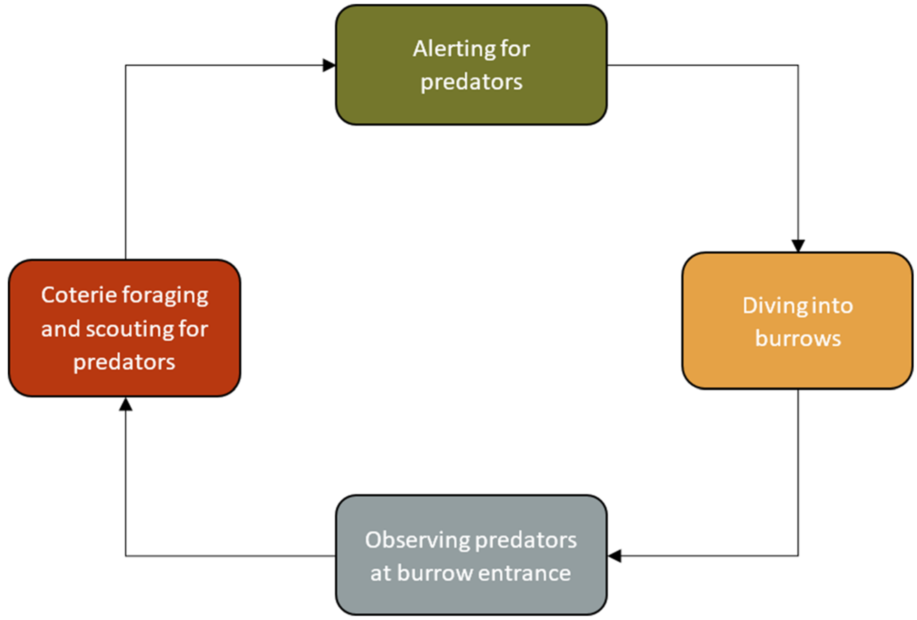
where r introduces the stochastic property to ensure exploration and takes the value -1 or 1 depending on the current iteration, alternating between -1 and 1 when the current iteration is odd or even, Δ represents a small number that accounts for differences that exist in the prairie dogs, though PDO implementation assumed they are all the same. $iter$ is the current iteration and Max_{iter} is the maximum number of iterations.

3.5 Exploitation

In this section, the exploitative behaviour of PDO is introduced. The specific response of the prairie dogs to two unique communication or alert sound, as shown in Fig. 4, is used to accomplish exploitation in the proposed PDO. The prairie dogs have signals or sounds for different scenarios ranging from predator threats to food availability. Their communication skills play a significant role in satisfying the prairie dogs' nutritional needs and anti-predation abilities. Furthermore, they can communicate the difference between the quality of food sources and predators and their hunting patterns. They equally have a unique response to these different sounds; for instance, if the communication denotes a quality food source, they converge at the sound source to meet their nutritional needs (feed). Also, if the communication species a hawk as the predator, only prairie dogs on the flight path of the hawk go into hiding, while others watch from their burrows.

These two specific behaviour results in the prairie dogs converging to a specific location or a promising location in the case of PDO implementation, where further search (exploitation) is carried out to find better or near-optimal solutions. The goal of the exploitation mechanisms used in PDO is to search the promising regions identified during the exploration phase intensively. The two strategies adopted for this phase are modelled in Eqs. 11 and 12. As mentioned earlier, the PDO moves between these two strategies subject to $\frac{Max_{iter}}{2} \le iter < 3\frac{Max_{iter}}{4}$ and $3\frac{Max_{iter}}{4} \le iter \le Max_{iter}$, respectively.

Fig. 4 Model of the exploitation



$$PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \varepsilon - CPD_{i,j} \times rand \, \forall \, \frac{Max_{iter}}{2} \leq iter < 3\frac{Max_{iter}}{4} \tag{11}$$

$$PD_{i+1,j+1} = GBest_{i,j} \times PE \times rand \, \forall \, 3\frac{Max_{iter}}{4} \leq iter < Max_{iter} \tag{12}$$

where $GBest_{i,j}$ is the global best-obtained solution so far, $eCBest_{i,j}$ evaluates the effect of the current obtained best solution as shown in Eq. 8, ε is a small number that represents the food source quality, $CPD_{i,j}$ is the cumulative effect of all prairie dogs in the colony and is defined in Eq. 8. PE represents the predator effect defined by Eq. 13, and rand denotes a random number between 0 and 1.

$$PE = 1.5 \times \left(1 - \frac{iter}{Max_{iter}}\right)^{\left(2\frac{iter}{Max_{iter}}\right)} \tag{13}$$

where $iter$ is the current iteration and Max_{iter} is the maximum number of iterations.

3.6 PDO pseudo-code

The pseudo-code of the optimization process of the PDO is given in Algorithm 1. PDO starts by generating a uniformly distributed random set of candidate solutions. After which, the algorithm repetitively uses its defined processes to explore all the possible positions of near-optimal solutions. Each time the algorithm finds the best solution so far and replaces the previously found solution based on the defined rule. The proposed algorithm uses four prairie dog activities to achieve exploration and exploitation. When$iter < \frac{Max_{iter}}{2}$, the PDO enters the exploration phase, while it enters the exploitation phase when $ter > \frac{Max_{iter}}{2}$. Finally, the proposed algorithm stops when the end criterion is met. Many stoppage criteria have been used in the literature; function tolerance, maximum execution time, and the maximum number of iterations are some examples. This study uses the maximum number of iterations as the PDO algorithm termination criteria. However, the choice of the maximum number of iterations needs to be carefully made because if it is too large or small, it will possibly lead to a waste of computing resources and a suboptimal solution, respectively. The flow chart shown in Fig. 5 gives an intuitive and detailed optimization procedure of the proposed PDO algorithm.

Algorithm 1. pseudo code of the optimization process of the PDO

Initialization
Set the PDO parameters: n, m, ρ, ε
Set $GBest$ and $CBest$ as $\emptyset$
Initialize the candidate solutions CT and PD
While $iter < Max_{iter}$ **do**
  For (i=1 to m) **do**
    For (j=1 to n) **do**
      Calculate the fitness of PD
      Find the Best solution so far (CBest)
      Update GBest
      Update DS and PE using Equations 9 and 12
      Update $CPD_{i,j}$ using Equation 8.
      If ($iter < \frac{Max_{iter}}{4}$) **then** **{foraging activities}**
        $PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \rho - CPD_{i,j} \times Lévy(n)$
      Else if ($\frac{Max_{iter}}{4} \leq iter < \frac{Max_{iter}}{2}$) **then** **{burrowing activities}**
        $PD_{i+1,j+1} = GBest_{i,j} \times eCBest_{i,j} \times DS \times Lévy(n)$
      Else if ($\frac{Max_{iter}}{2} \leq iter < 3\frac{Max_{iter}}{4}$) **then {food alarm}**
        $PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \varepsilon - CPD_{i,j} \times rand$
      Else **{antipredation alarm}**
        $PD_{i+1,j+1} = GBest_{i,j} \times PE \times rand$
      End If
    End For
  End For
  $iter = iter + 1$
End While
Return *best solution* **(GBest)**
End

3.7 Computational complexity

The proposed PDO's computational complexity is influenced by initialization processes, the number of fitness function evaluations, and updating of positions of the solutions. The computational complexity of initializing m candidate solutions is $O(m)$. The updating position process has a computational complexity of $O(Max_{iter} \times m)n$, and the complexity of the function evaluation is dependent on the optimization problem, and it is given as $O(\text{Obj})$. Therefore, the computational complexity of the proposed PDO is given in Eq. 14.

$$O(PDO) = O(mnMax_{iter}) \times O(Obj) \quad (14)$$

where Max_{iter} is the number of iterations, m represents the number of candidate solutions, and n presents the number of prairie dogs in a coterie.

3.8 Conceptual comparative advantages of PDO

Source of inspiration and choice of position updating mechanism are the two features that differentiate one metaheuristic algorithm from another. Typically, the matrix of randomly generated solutions within the problem boundary or constraints is constructed for the optimization problem under consideration [58]. The constructed matrix is repeatedly updated as the algorithm iterates through the updating mechanism. The PDO also initiates an optimization process just like all other population-based metaheuristic algorithms by random movement of search agents in the problem search space; however, some of the significant differences are described as follows:

i. The PDO divides the problem search space into several coteries, and the optimization process is carried out in each of the coteries. The solutions are

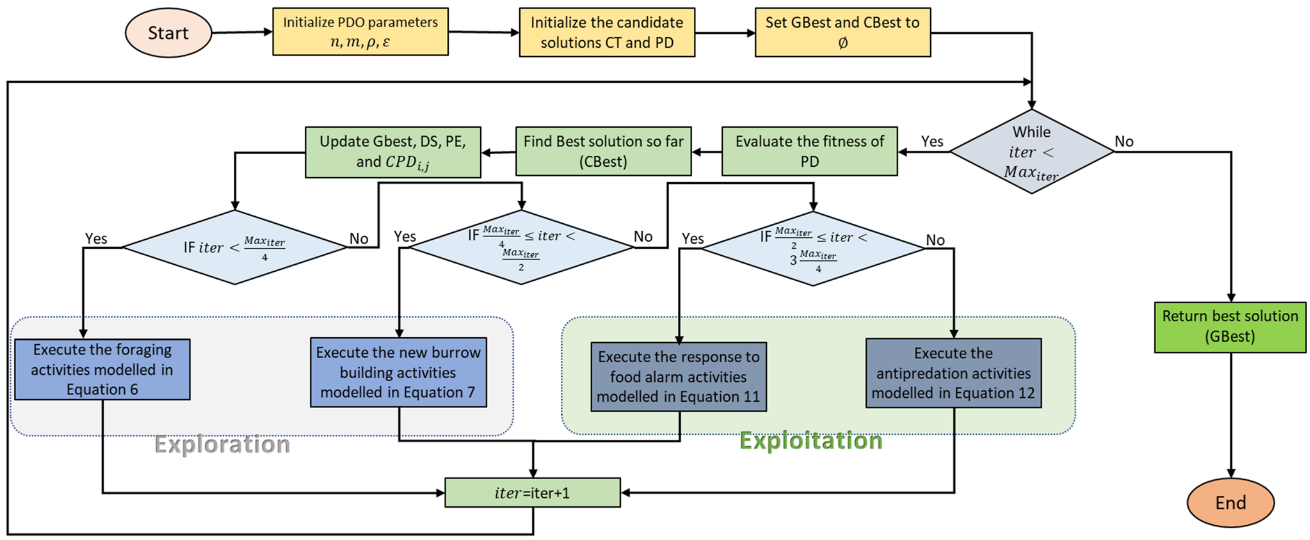


Fig. 5 Flow chart of the proposed PDO algorithm

aggregated, and the best is obtained from the aggregate.

ii. The foraging and burrow building activities (exploration), communication, and anti-predation (exploitation) activities are uniquely modelled as in Eqs. 6, 7, 11, and 12 with the introduction of the digging strength (DS) and predator effect (PE) attributes that uniquely affect the updating process of PDO.

iii. The PDO effectively utilizes division of labour, with each coterie effectively performing optimization tasks within its region or boundary.

4 Simulation, results, and discussion

The newly proposed metaheuristic method PDO is applied to solve different global optimization problems, comprising twenty-two classical and ten CEC 2020 benchmark test functions. The PDO is also applied to solve twelve engineering problems. The results of PDO are compared with arithmetic optimization algorithm (AOA) [59], constriction coefficient-based (PSO) and GSA (CPSOGSA) [60], particle swarm optimization (PSO) [11], biogeography-based optimization (BBO) [61], differential evolution (DE) [29], salp swarm algorithm (SSA) [62], sine cosine algorithm (SCA) [63], grey wolf optimizer (GWO) [64], and dwarf mongoose optimization (DMO) [58].

The documentation, choice of optimization problems, and experimental conditions of most metaheuristic algorithms available in the literature are sometimes not detailed enough to allow for exact replication. As such, in this study, the performance and availability of codes influenced our choice of algorithms. In most cases, the basic version of the algorithms is used for comparative analysis. The maximum number of iterations is set to 1000, and the population size is set to 50 for all algorithms considered in this study. Furthermore, the statistical measures are computed for each algorithm after 30 runs, and four performance indicators are used to validate the effectiveness of the proposed PDO in conjunction with other state-of-the-art optimization algorithms. The performance metrics used include Best, Worst, Average, and Standard deviation (SD). Also, the parameter setting of each algorithm is given in Table 1. The experiment was conducted using 64-bit Windows 10 OS, Intel Core i7-7700@3.60 GHz CPU, 16G RAM, and the PDO source code is implemented using MATLAB (R2020b). Moreover, a nonparametric statistical test called the Friedman ranking test is applied for a fair comparison with other existing optimization methods.

4.1 Benchmark test function

In this subsection, twenty-two (22) classical test functions and ten CEC 2020 are chosen as case studies. The mathematical formulations of the test functions are presented in Tables 2 and 3, respectively. The classical test functions are classified into three categories; the unimodal functions (F1–F7) have a single global optimum solution in the search domain. The multimodal (F8–F13) and multimodal with fixed dimension (F14–F22) functions have more than one global optimum solution in the search space. The CEC 2020 test suite consists of 10 functions that are designed specially to make finding the global optimum difficult.

The problem landscape perspective of some selected classical benchmark functions is shown in Fig. 6. The representative test functions include F1–10, F12–F14, and F21, respectively. It may be observed in the shapes

Table 1 Algorithm control parameters

Algorithm	Parameter	Value
PDO	Food source alarm ρ	0.1
	Food source quality ε	2.2204e−16
	Individual PD difference Δ	0.005
DMO	Number of babysitters	3
	Alpha female vocalization	2
AOA	α	5
	μ	0.05
PSO	C_1, C_2 (personal and social constants)	2
	Wmax (maximum inertia weight)	0.9
	Wmin (minimum inertia weight)	0.2
CPSOGSA	$< p1, < f > 2$ (control parameters)	2.05
GWO	a (area vector)	[0, 2]
	r_{1f} r_2 (random vectors)	[0, 1]
SCA	a (constant)	2
SSA	c_2, c_3 (random numbers)	[0, 1]
BBO	nKeep (number of habitats retained after every generation)	0.2
	Pmutation (mutation probability)	0.9
DE	Lower bound of scaling factor	0.2
	Upper bound of scaling factor	0.8
	PCR (crossover probability)	0.8

presented in Fig. 6 that the class of unimodal test functions has only one optimum with no local optima. Meanwhile, the implication for this class of function is that their search spaces are appropriate for testing the convergence speed and exploitive behaviour of the new PDO algorithm compared to other state-of-the-art related methods. On the one hand, the multimodal and composite benchmark test functions have many optima, making them more suitable for benchmarking any optimization algorithms' local optima avoidance and explorative behaviour [62]. On the other hand, the composite test functions are usually more difficult to solve than their multi-modal counterparts. One advantage of the composite benchmark function is its high similarity to real problem search spaces.

4.1.1 Comparison of PDO with other metaheuristic algorithms on benchmark functions

In this sub-section, the performance of the PDO to find the optimal solution for classical functions is evaluated. Tables 4, 5 and 6 show the comparative results of PDO and ten other metaheuristic algorithms. Our choice of algorithms is based on the availability of code online and their established abilities to solve these kinds of global optimization problems. Setting the dimension of the high-dimensional functions (F1–F13) to 10, the results are shown in Table 4. The PDO ranked first among other algorithms considered, closely followed by DE and PSO, which ranked second and third. Moreover, the PDO has the smallest fitness value average at seventeen functions, representing nearly 77.3%.

Similarly, the performance of PDO was tested by varying the dimension of F1–F13 to 100 and 1000, as shown in Tables 5 and 6, respectively. It is evident in the tables that the PDO returned the least average value of fitness function at these dimensions, which has the first rank. The GOA ranked second for both dimensions. Analysing the behaviours of the PDO and GOA at this dimension, PDO is ranked first on eight out of the thirteen functions, whereas the GOA ranked first in four functions only. Overall, the performance of PDO at most of the functions is competitive. The PDO maintained steady performance across these varied dimensions, meaning that PDO could perform equally well in different real-world applications that depend on high dimensions.

The convergence curve of the PDO and other algorithms used in this study is shown in Fig. 7 (F1–F22). It can be seen from the convergence curves that the PDO has a faster convergence rate in most functions than other algorithms for the tested functions. Overall, PDO's convergence is competitive, and the PDO needs fewer iterations to reach the optimal value in most tested functions.

4.1.2 Comparison of PDO with other metaheuristic algorithms using the CEC 2020 benchmark suite

In this sub-section, the performance of the PDO is to find the optimal solution for complex functions in the CEC

Table 2 Classical test functions

ID	Type	Function	Dimension	Bounds	Global
F1	Unimodal	$f(x) = \sum_{i=1}^{n} x_i^2$	10,100,1000	[− 100,100]	0
F2	Unimodal	$f(x) = \sum_{i=0}^{n} \lvert x_i\rvert + \prod_{i=0}^{n} \lvert x_i\rvert$	10,100,1000	[− 10,10]	0
F3	Unimodal	$f(x) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} x_j\right)^2$	10,100,1000	[− 100,100]	0
F4	Unimodal	$f(x) = max_i\{\lvert x_i\rvert, 1 \le i \le n\}$	10,100,1000	[− 100,100]	0
F5	Unimodal	$f(x) = \sum_{i=1}^{n-1} [100(x_i - x_{i+1})^2 + (1 - x_i)^2]$	10,100,1000	[− 30,30]	0
F6	Unimodal	$f(x) = \sum_{i=1}^{n} ([x_i - 0.5])^2$	10,100,1000	[− 100,100]	0
F7	Unimodal	$f(x) = \sum_{i=0}^{n} ix_1^4 + rand[0,1)$	10,100,1000	[− 128,128]	0
F8	Multimodal	$f(x) = \sum_{i=1}^{n} -x_i sin\left(\sqrt{\lvert x_i\rvert}\right)$	10,100,1000	[− 500,500]	− 418.9829 × n
F9	Multimodal	$f(x) = 10 + \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i))$	10,100,1000	[− 5.12,5.12]	0
F10	Multimodal	$f(x) = -aexp\left(-0.02\sqrt{n^{-1}\sum_{i=1}^{n} x_i^2}\right) - exp\left(n^{-1}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + a + e, a = 20$	10,100,1000	[− 32,32]	0
F11	Multimodal	$f(X) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos\left(\frac{x_i}{\sqrt{i}}\right)$	10,100,1000	[− 600,600]	0
F12	Multimodal	$f(x) = \frac{\pi}{n}\{10\sin(\pi y_i)\} + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10sin^2(\pi y_{i+1}) + \sum_{i=1}^{n} u(x_i, 10, 100, 4)]$	10,100,1000	[− 50,50]	0
		Where $y_i = 1 + \frac{x_i+1}{4}, u(x_i, a, k, m)\begin{cases} K(x_i - a)^m if x_i > a \\ 0 - a \le x_i \ge a \\ K(-x_i - a)^m - a \le x_i \end{cases}$			
F13	Multimodal	$f(x) = 0.1(\sin^2(3\pi x_1) + \sum_{i=1}^{n} (x_i - 1)^2 [1 + sin^2(3\pi x_i$ $+1)] + (x_n - 1)^2 1 + sin^2(2\pi x_n)) + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	10,100,1000	[− 50,50]	0
F14	Fixed-dimension multimodal	$f(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^6}\right)^{-1}$	2	[− 65,65]	1
F15	Fixed-dimension multimodal	$f(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	[− 5,5]	0.00030
F16	Fixed-dimension multimodal	$f(x) = ax_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[− 5,5]	− 1.0316
F17	Fixed-dimension multimodal	$f(X) = \left[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1{}^2 - 14x_2 + 6x_1x_2 + 3x_2{}^2)\right]$ $\times\left[30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1{}^2 + 48x_2 - 36x_1x_2 + 27x_2{}^2\right]$	2	[− 2,2]	3
F18	Fixed-dimension multimodal	$f(x) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{i=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3	[− 1,2]	− 3.86
F19	Fixed-dimension multimodal	$f(x) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{i=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6	[1, 1]	− 3.2
F20	Fixed-dimension multimodal	$f(x) = -\sum_{i=1}^{5} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[1, 1]	− 10.1532
F21	Fixed-dimension multimodal	$f(x) = -\sum_{i=1}^{7} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[1, 1]	− 10.4028
F22	Fixed-dimension multimodal	$f(x) = -\sum_{i=1}^{10} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[1, 1]	− 10.5363

Table 3 CEC 2020 test functions

Type	Number	Functions	F_i^*
Unimodal function	1	Shifted and Rotated Bent Cigar Function (CEC 2017[4] F1)	100
Basic functions	2	Shifted and Rotated Schwefel's Function (CEC 2014[3] F11)	1100
	3	Shifted and Rotated Lunacek bi-Rastrigin Function (CEC 2017[4] F7)	700
	4	Expanded Rosenbrock's plus Griewank's Function (CEC2017[4] f19)	1900
Hybrid functions	5	Hybrid Function 1 (N = 3) (CEC 2014[3] F17)	1700
	6	Hybrid Function 2 (N = 4) (CEC 2017[4] F16)	1600
	7	Hybrid Function 3 (N = 5) (CEC 2014[3] F21)	2100
Composition functions	8	Composition Function 1 (N = 3) (CEC 2017[4] F22)	2200
	9	Composition Function 2 (N = 4) (CEC 2017[4] F24)	2400
	10	Composition Function 3 (N = 5) (CEC 2017[4] F25)	2500

*Search range: $[-100,100]^D$

2020 test suit. The comparative results of PDO and ten other algorithms are shown in Table 7. The PDO outperformed the competitive algorithms used in this study, and the PDO ranked first among the algorithms. PSO and DE closely follow the PDO with a mean rank of 3.3 and 3.75, respectively. The average fitness value returned by the PDO is the least among all the algorithms, indicating the PDO's stability in solving these problems.

Furthermore, the convergence curve of the competitive algorithms is shown in Fig. 7 (F23–F32). Just like in the classical functions, the PDO showed fast convergence for most of the functions in the CEC 2020, and it was competitive in the others. The ability of the PDO to find better solutions and convergence faster than the other competitive algorithms confirms the stability of PDO in terms of exploitation and exploration. The contributory features of PDO that enable this performance are as follows:

- The values taken by the DS and PE as defined in Eqs. 10 and 13 introduce a stochastic scaling factor that increases the diversity of the solution after each iteration to escape local optima and encourages fast convergence.
- The aggregation of results of each coterie ensures better decisions on the best-so-far value through historical learning.

4.1.3 Visual observation of PDO's performance

The qualitative results evaluation of the proposed PDO algorithm on seven classical and multimodal standard benchmark functions, F1, F3, F5, F7, F9, F11, and F13, is shown in Fig. 8. The qualitative analysis of the PDO results was conducted based on the following performance metrics: 3D parameter search space map representation, search histories, search trajectory, the average fitness function of the search agent population, and the convergence curve characteristics. Moreover, the qualitative results analysis's essence is to determine how the PDO algorithm performs in generating good-quality solutions when tested on different mathematical benchmark test functions having variations in problem complexities, graph sizes, and dimensions. Another perspective to this type of qualitative result analysis is investigating whether the PDO can retain its search scalability features when tested on problems with large graph sizes.

By analysing the individual qualitative results analysis metrics, the 3D maps show the problem landscape of the selected test functions. On the other hand, the search history points out the various location history of all search agents during the optimization process. In the second column of Fig. 8, we can observe the sampled regions of the search space by the PDO and its probable search patterns in the entire population. In the third column depicting the search agent trajectory of the PDO, the graphs reveal the dynamism in how the PDO can transform from sudden to gradual changes in the initial steps of the iteration process within the problem search space and with a repetitive characteristic in the final stages. This shows that the PDO first requires its search agents or prairie dogs to explore the search space adequately. Afterwards, the PDO exploits the search space by moving the agents closer to the location of the global optimum, thereby encouraging the agents to converge locally instead of globally.

The fourth column of Fig. 8, illustrating the average fitness function, shows how the average fitness of the entire population changes at every optimization step. Furthermore, the fitness curves reveal how well the PDO algorithm improves the population during the exploration and exploitation optimization process. Finally, looking at the

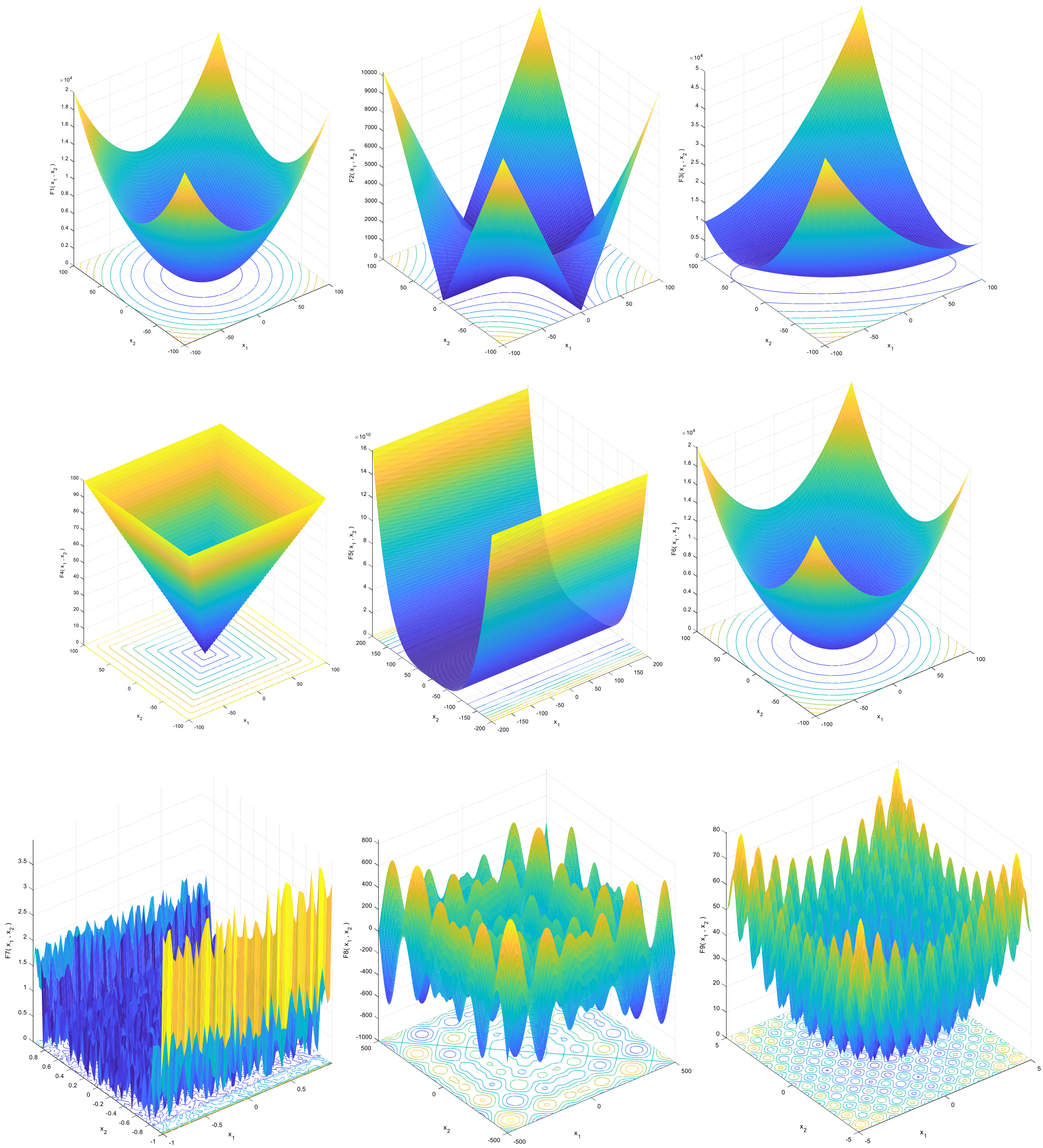


Fig. 6 3D perspective view of selected functions, namely F1–10, F12–F14, and F21 for PDO implementation

convergence curve obtained by the PDO, as shown in Fig. 8, it can be observed that the algorithm improved the fitness of the approximation of the global optimum obtained in each of the iteration steps throughout the entire optimization process. However, this improvement is not always consistent, as seen in some of the figures. This demonstrates that the new PDO algorithm behaves differently on different problems.

Moreover, the recorded improvement pattern exhibited by the PDO over the population is considered not to be always consistent, as can be seen in some of the presented convergence curves. This means that the PDO exhibits different convergence characteristics or patterns depending

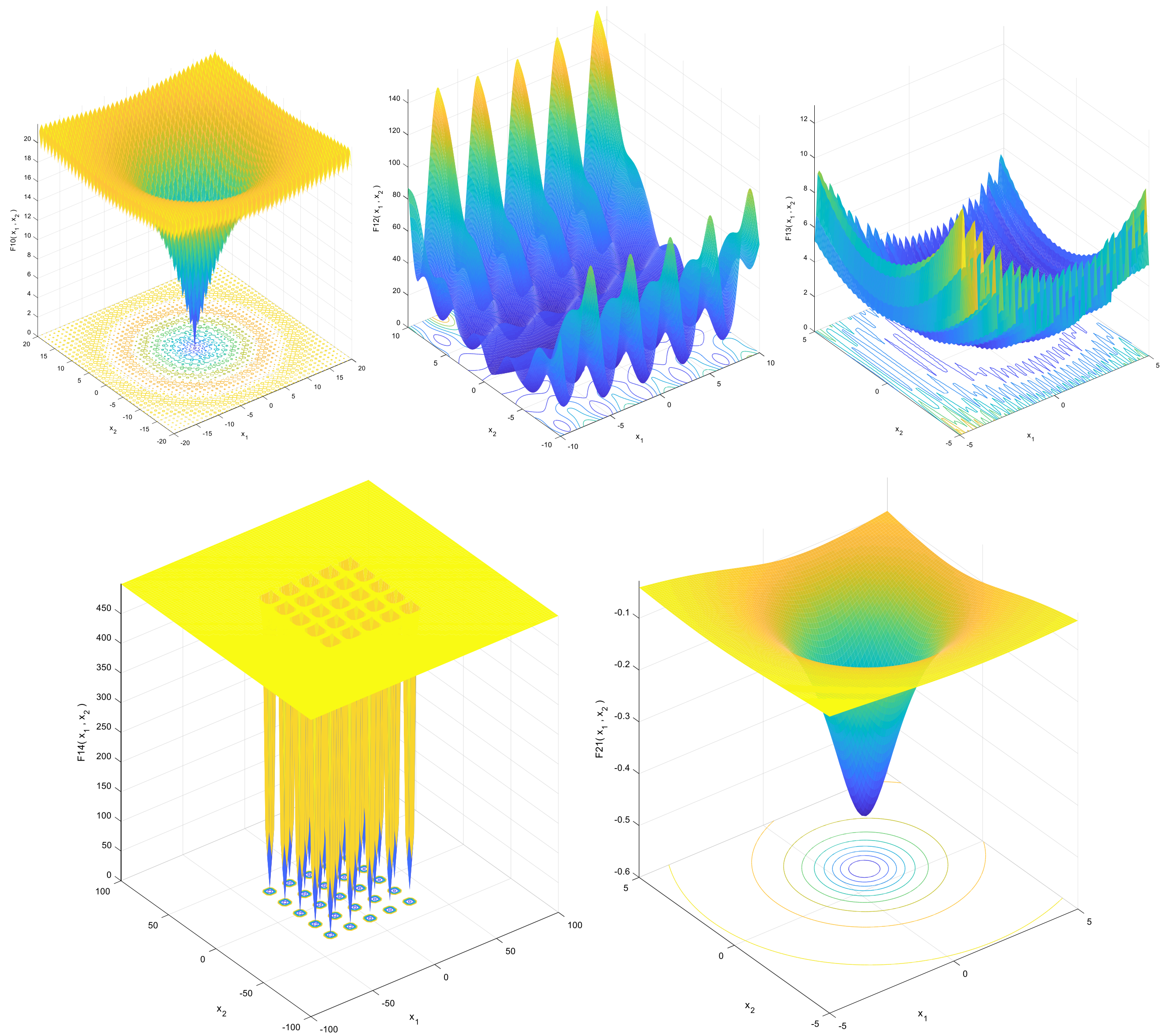


Fig. 6 continued

on the problem level of complexity and features. Figure 8 demonstrates the evaluation of the qualitative results analysis of PDO on classical and multimodal standard benchmark test functions.

4.2 Real-world application

The PDO was used to solve twelve constraint real-world engineering optimization problems, and the definition of these problems is given in subsequent subsections. The parameters for the competitive algorithms are the same as the previous experiments for classical and CEC 2020 test functions. Similar to the previous experimentation conducted using mathematical test functions, for the real-world engineering design problem, the maximum number of iterations is set to 1000, and the population size is set to 50 for all algorithms considered in this study. Furthermore, the statistical measures are computed for each algorithm after 30 runs, and four performance indicators are used: Best, Worst, Average, and SD. Also, the parameter setting of each algorithm is given in Table 1.

4.2.1 The welded beam design problem

The welded beam design (WBD) problem is used to test the performance of PDO. The schematic diagram of the welded beam is shown in Fig. 9, and the WBD is subjected to four design constraints, namely, shear (τ) and beam blending (θ) stress, bar buckling load (P_c) beam end deflection (δ). Finding the minimum manufacturing cost of the design

Table 4 Comparative results for (F1–F13) set to 10 and other fixed dimensional functions (F14–F22)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F1	10	0	Best	0	0	0	0	0	0	2.27E−08	0	0	0	0
			Worst	0	0	0	0	0	0	8.33E−06	0	0	0	0
			Average	0	0	0	0	0	0	1.1E−06	0	0	0	0
			SD	0	0	0	0	0	0	1.75E−06	0	0	0	0
			Rank	1	1	1	1	1	1	2	1	1	1	1
F2	10	0	Best	0	0	0	0	0	0	1.24E−05	0	3.29E−06	0	0
			Worst	0	0	0	0	0	0	0.000201	0	1.88E−05	0	0
			Average	0	0	0	0	0	0	7.61E−05	0	7.19E−06	0	0
			SD	0	0	0	0	0	0	4.81E−05	0	3.11E−06	0	0
			Rank	1	1	1	1	1	1	3	1	2	1	1
F3	10	0	Best	0	0	0	0	0	0	0.001819	0.058607	0	0	0
			Worst	0	0	0	0	0	0	0.020846	0.44565	0	0	0
			Average	0	0	0	0	0	0	0.007805	0.20423	0	0	0
			SD	0	0	0	0	0	0	0.004558	0.10584	0	0	0
			Rank	1	1	1	1	1	1	2	3	1	1	1
F4	10	0	Best	0	0	0	0	0	0	0.002741	5.51E−08	9.43E−06	0	0
			Worst	0	0	0	0	0	0	0.017679	3.26E−07	1.85E−05	1.43E−08	0
			Average	0	0	0	0	0	0	0.007329	1.64E−07	1.33E−05	4.75E−10	0
			SD	0	0	0	0	0	0	0.003554	7.93E−08	2.32E−06	2.6E−09	0
			Rank	1	1	1	1	1	1	4	2	3	1	1
F5	10	0	Best	2.62E−05	5.5019	0	4.6732	0.20837	0.25725	0.042267	0.09975	4.2179	6.1717	5.0595
			Worst	0.001631	8.7312	8.9902	5.8353	262.2	4.5866	15.888	6.7783	386.3	8.0797	7.2023
			Average	0.000237	7.1658	7.0876	5.1486	34.932	3.4081	5.9404	1.7159	40.951	6.9003	6.0783
			SD	0.000306	0.91571	3.6114	0.28308	74.046	1.2584	4.0684	1.8038	85.319	0.45096	0.55493
			Rank	1	8	8	4	9	3	5	2	10	7	6
F6	10	0	Best	0	0.002008	1.3943	0.0048	0	0	1.43E−08	0	0	0.11612	3.69E−07
			Worst	0	0.77724	2.25	0.022207	0	0	1.95E−06	0	0	0.53614	1.15E−06
			Average	0	0.22968	1.8804	0.012455	0	0	5.73E−07	0	0	0.27222	6.56E−07
			SD	0	0.2326	0.18309	0.004317	0	0	5.7E−07	0	0	0.10582	2.15E−07
			Rank	1	5	7	4	1	1	2	1	1	6	3
F7	10	0	Best	1E−06	8.75E−07	4.38E−07	1.14E−07	1.46E−06	1.6E−06	2.75E−07	1.37E−07	9.43E−07	3.16E−06	1.9E−06
			Worst	2.21E−05	0.000113	5.25E−05	7.89E−05	5.09E−05	6.91E−05	7.35E−05	6.74E−05	0.000117	0.000101	4.98E−05
			Average	8.01E−06	2.69E−05	1.49E−05	2.53E−05	1.93E−05	1.96E−05	2.14E−05	1.98E−05	2.63E−05	3.14E−05	1.8E−05
			SD	4.95E−06	2.92E−05	1.09E−05	2.33E−05	1.45E−05	1.59E−05	2.14E−05	1.74E−05	2.79E−05	2.59E−05	1.35E−05
			Rank	1	9	2	8	4	5	7	6	10	11	3

Table 4 (continued)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F8	10	−4189.8	Best	−4189.7	−3551.7	−2392.2	−4071.4	−3558.1	−3479.2	−3972.7	−4189.8	−3498.9	−2736.2	−3293.1
			Worst	−4020.9	−2455.3	−1854.9	−2847.5	−2194.5	−2808	−2966	−4189.8	−2209.6	−2019.5	−2228.1
			Average	−4147.5	−2876.8	−2126	−3530.5	−2886	−3111.8	−3529.8	−4189.8	−2903.9	−2320.9	−2733.9
			SD	57.212	279.42	120.6	272.09	348.4	194.69	285.29	2.78E−12	349.72	171.82	264.15
			Rank	2	6	3	11	7	9	10	1	8	4	5
F9	10	0	Best	0	0	0	0	6.9647	2.9857	2.9849	0	3.9798	0	0
			Worst	0	15.395	0	0	48.753	16.914	13.929	0	26.864	15.589	6.2667
			Average	0	1.913	0	0	20.43	10.215	6.7989	0	14.128	0.91261	0.20889
			SD	0	3.9178	0	0	10.122	3.7495	2.8752	0	7.3226	3.5087	1.1441
			Rank	1	2	1	1	8	6	4	1	7	5	3
F10	10	0	Best	0	0	0	0	0	0	0.000104	0	6.4E−06	0	0
			Worst	0	3.197	0	0	0	0	0.000719	0	2.0133	0	0
			Average	0	0.10662	0	0	0	0	0.000301	0	0.53401	0	0
			SD	0	0.58369	0	0	0	0	0.000167	0	0.73626	0	0
			Rank	1	3	1	1	1	1	2	1	4	1	1
F11	10	0	Best	0	0	0	0	0.044258	0.012316	0.007398	0	0.046707	0	0
			Worst	0.014773	0.23598	0	0	0.74097	0.15251	0.083613	0	0.55523	0.44807	0.054735
			Average	0.000492	0.01984	0	0	0.26065	0.071743	0.0472	0	0.23162	0.02325	0.01502
			SD	0.002697	0.044677	0	0	0.18155	0.033792	0.020439	0	0.12381	0.083556	0.018243
			Rank	2	4	1	1	9	7	6	1	8	5	3
F12	10	0	Best	0	0.000395	0.061063	0.000217	0	0	0	0	0	0.011676	6.49E−08
			Worst	0	0.16786	2.4347	0.63623	1.8684	0	2.96E−07	0	1.244	0.17761	0.020027
			Average	0	0.03538	0.46583	0.023788	0.30086	0	2.3E−08	0	0.062201	0.062388	0.001251
			SD	0	0.038961	0.44594	0.11573	0.53298	0	6.46E−08	0	0.23671	0.037729	0.004772
			Rank	1	4	8	5	0	1	2	1	6	7	3
F13	10	0	Best	0	0.001955	0	0.46975	0	0	0	0	0	0.15457	1.95E−07
			Worst	1.31E−08	0.63082	0.9	0.99431	0	0	1.81E−07	0	0.010987	0.47033	0.10011
			Average	8.03E−10	0.29821	0.09	0.82108	0	0	3.61E−08	0	0.000366	0.2681	0.006644
			SD	3.07E−09	0.14465	0.27462	0.14167	0	0	3.79E−08	0	0.002006	0.078078	0.025283
			Rank	2	7	6	9	1	1	3	1	4	8	5
F14	2	1	Best	0.998	0.998	0.998	2.9821	0.998	0.998	0.998	0.998	0.998	0.998	0.998
			Worst	0.998	4.9505	1.0057	12.671	20.153	0.998	15.504	0.998	0.998	2.9821	10.763
			Average	0.998	1.9895	0.99858	10.561	5.3308	0.998	4.8894	0.998	0.998	1.1965	3.2208
			SD	0	1.1342	0.001822	3.1698	4.7139	0	3.798	0	2.16E−16	0.60539	3.5429
			Rank	1	5	3	9	8	1	7	1	2	4	6

Table 4 (continued)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F15	0	0.0003	Best	0.000307	0.000308	0.000694	0.000327	0.000355	0.000307	0.00031	0.000479	0.000317	0.000308	0.000307
			Worst	0.000307	0.001606	0.001512	0.10929	0.020363	0.020363	0.020363	0.00076	0.020364	0.001419	0.020363
			Average	0.000307	0.000522	0.000965	0.010701	0.005947	0.002087	0.002725	0.000665	0.001554	0.000957	0.004384
			SD	1.94E−13	0.000324	0.000341	0.021268	0.008843	0.004994	0.005989	7.58E−05	0.003567	0.000377	0.00813
			Rank	1	2	5	11	10	7	8	3	6	4	9
F16	2	−1.0316	Best	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316
			Worst	− 1.0316	− 1.0316	− 1.0315	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316
			Average	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316
			SD	6.39E−16	2.55E−07	3.12E−05	6.67E−08	6.32E−16	6.78E−16	1.24E−15	6.78E−16	4.76E−15	1.69E−05	2.78E−09
			Rank	2	8	10	7	1	3	4	3	5	9	6
F17	2	3	Best	3	3	3	3	3	3	3	3	3	3	3
			Worst	3	3.001	3.0001	30	3	3	30	3	3	3	3
			Average	3	3.0002	3	11.1	3	3	3.9	3	3	3	3
			SD	8.33E−16	0.000259	2.69E−05	12.584	1.26E−15	6.23E−16	4.9295	1.97E−15	6.89E−14	3.62E−06	4.45E−06
			Rank	2	9	8	11	3	1	10	4	5	7	6
F18	3	− 3.86	Best	− 3.8628	− 3.8628	− 3.8617	− 3.8608	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.862	− 3.8628
			Worst	− 3.8628	− 3.8547	− 3.8544	− 3.8498	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.8526	− 3.8549
			Average	− 3.8628	− 3.8608	− 3.8555	− 3.854	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.8628	− 3.8559	− 3.8623
			SD	2.71E−15	0.003022	0.002082	0.002684	2.54E−15	2.71E−15	2.26E−15	2.71E−15	2.2E−14	0.002778	0.001892
			Rank	1	6	3	4	1	1	1	1	1	5	2
F19	6	− 3.32	Best	− 3.322	− 3.3208	− 3.1652	− 3.2102	− 3.322	− 3.322	− 3.322	− 3.322	− 3.322	− 3.2088	− 3.322
			Worst	− 3.322	− 2.8376	− 1.9177	− 3.0065	− 3.2031	− 3.2031	− 3.2031	− 3.322	− 3.1987	− 2.2567	− 3.0207
			Average	− 3.322	− 3.2445	− 3.0569	− 3.1086	− 3.219	− 3.2625	− 3.2943	− 3.322	− 3.2224	− 3.0057	− 3.2622
			SD	2.28E−11	0.12036	0.24041	0.055956	0.041107	0.060463	0.051146	1.43E−15	0.045329	0.1696	0.088271
			Rank	1	7	10	9	2	5	4	1	3	8	6
F20	4	− 10.1532	Best	− 10.153	− 9.9702	− 8.7714	− 7.5309	− 10.153	− 10.153	− 10.153	− 10.153	− 10.153	− 6.506	− 10.153
			Worst	− 10.153	− 2.5645	− 0.88199	− 2.2566	− 2.6305	− 2.6305	− 2.6305	− 10.131	− 2.6305	− 0.49729	− 5.0552
			Average	− 10.153	− 5.5877	− 5.5321	− 4.2697	− 6.4731	− 7.4761	− 5.4851	− 10.152	− 8.7275	− 3.6039	− 8.9694
			SD	5.41E−15	2.849	1.4164	1.0722	3.3954	3.4128	3.4524	0.00406	2.6947	2.0591	2.1819
			Rank	1	8	9	10	4	5	3	2	6	11	7
F21	4	− 10.4028	Best	− 10.403	− 10.376	− 8.4361	− 7.2375	− 10.403	− 10.403	− 10.403	− 10.403	− 10.403	− 8.4693	− 10.403
			Worst	− 10.403	− 2.6892	− 0.90808	− 1.7517	− 1.8376	− 2.7519	− 2.7519	− 10.403	− 2.7519	− 0.90968	− 5.0877
			Average	− 10.403	− 5.8629	− 5.9236	− 4.1539	− 7.2929	− 8.6769	− 7.3087	− 10.403	− 9.6191	− 4.6177	− 10.225
			SD	3.63E−15	2.9202	1.4845	1.2648	3.6856	2.9752	3.4262	1.62E−08	2.0677	1.733	0.97036
			Rank	1	7	9	10	6	4	5	1	3	8	2

Table 4 (continued)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F22	4	− 10.5363	Best	− 10.536	− 10.393	− 10.118	− 7.8174	− 10.536	− 10.536	− 10.536	− 10.536	− 10.536	− 9.4044	− 10.536
			Worst	− 10.536	− 2.4055	− 5.1285	− 2.1909	− 2.4217	− 2.4217	− 1.8595	− 10.536	− 5.1285	− 0.94553	− 2.4217
			Average	− 10.536	− 5.7166	− 6.371	− 4.4501	− 7.0219	− 8.8234	− 7.1483	− 10.536	− 9.9988	− 5.1252	− 9.8149
			SD	3.76E−14	2.5849	1.3998	1.7161	3.8783	3.2147	3.7315	2.6E−15	1.6405	1.9278	2.2386
			Rank	1	7	8	10	6	4	5	1	2	9	3
Mean rank				3.02	7.20	6.55	7.25	6.55	4.75	7.25	3.70	6.59	7.66	5.48
Rank				1	7	5	8	5	3	8	2	6	9	4

problem is the goal of the optimization problem [65]. The design variables for WBD are: $h = x_1, l = x_2, t = x_3, b = x_4$.

Where the l is the length, h is the height, t is the thickness, and b is the weld thickness of the bar.

The mathematical formulation of the WBD problem is given in Eq. 15.

$$\min f(X) = x_1^2 x_2 1.10471 + 0.04811 x_3 x_4 (14.0 + x_2) \tag{15}$$

Subject to

$$\begin{aligned} &s_1(X) = \tau(X) - \tau_{max} \le 0, \\ &s_2(X) = \sigma(X) - \sigma_{max} \le 0, s_3(X) = \delta(X) - \delta_{max} \le 0, \\ &s_4(X) = x_1 - x_4 \le 0, s_5(X) = P - P_c(X) \le 0, s_6(X) \\ &\quad = 0.125 - x_1 \le 0, \\ &s_7(X) = 1.10471 x_1^2 + 0.04811 x_3 x_4 (14.0 + x_2) - 5.0 \le 0 \end{aligned}$$

The interval for the design variables:

$$0.1 \le x_1 \le 2, 0.1 \le x_2 \le 10, 0.1 \le x_3 \le 10, 0.1 \le x_4 \le 2$$

where

$$\begin{aligned} &\tau\left(\vec{l}\right) = \sqrt{\tau'^2 + 2\tau'\tau''^{\left(\frac{x_2}{R}\right)} + (\tau'')^2}, \quad \tau' = P/\sqrt{2x_1x_2}, \\ &\quad \tau'' = MR/J, \quad M = P\left(L + \frac{x_2}{2}\right) \\ &J = 2\left\{\sqrt{2}x_1x_2\left[\left(\frac{(x_2^2)}{4}\right) + \left(x_1 + \frac{x_3}{2}\right)^2\right]\right\}, \\ &\quad P_c(X) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{E}/4G\right) \end{aligned}$$

The parameters for WBD are set as follows: $\sigma_{max} = 3000psi$, P = 6000lb, L = 14in, $\delta_{max} = 0.25in$, $E = 3 \times 10^6 psi$, $\tau_{max} =$ 13600psi, and $G = 12 \times 10^6 psi$ [65].

The performance of the proposed PDO in solving the WBD problem compared with other competitive algorithms is shown in Table 8. The proposed PDO obtains the smallest manufacturing cost, followed by the GWO. Also, one can depict the behaviour of PDO from the convergence curves and best positions shown in Fig. 10. The manufacturing cost value decreased steadily after the 20th iteration, indicating a balance between exploration and exploitation. This balance increased the ability of PDO to escape or avoid local optima.

4.2.2 The pressure vessel design problem (PVD)

The pressure vessel design (PVD) problem is one of the mixed-integer-type optimization problems found in the engineering domain. The goal of the PVD is to minimize the cost of raw materials and welding of the pressure

Table 5 Comparative results for (F1–F13) set to 100

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWOs
F1	100	0	Best	0	0	0	0.004654	187.57	4.8318	121.54	12.648	0.000136	42.265	0
			Worst	0	0	0	0.029063	10,059	74.542	190.06	28.817	0.050999	12,019	0
			Average	0	0	0	0.015986	1464	18.895	152.02	20.416	0.009704	3960.4	0
			SD	0	0	0	0.006413	2926.2	14.605	18.147	3.2124	0.012482	3281.6	0
			Rank	1	1	1	3	7	4	6	5	2	8	1
F2	100	0	Best	0	0	0	0	18.799	74.133	7.0335	1.8703	9.2635	0.0094545	0
			Worst	0	0	0	0	344.53	374.31	8.6771	2.5959	21.234	8.2428	0
			Average	0	0	0	0	125.67	279.78	7.7639	2.1907	14	1.6637	0
			SD	0	0	0	0	111.64	76.251	0.3968	0.18717	2.901	2.1935	0
			Rank	1	1	1	1	6	7	4	3	5	2	1
F3	100	0	Best	0	1.02E−05	1.09E−05	0.10209	48,231	43,229	13,147	292,790	7830.3	115,790	0.000169
			Worst	0	1234.5	3956.8	3.0422	132,820	79,941	32,739	429,180	49,585	281,850	6.6058
			Average	0	67.358	673.91	0.68079	88,847	56,144	21,465	383,050	22,728	179,440	0.67758
			SD	0	249.22	1004.2	0.77077	22,235	8636.8	3543.5	32,584	10,792	39,699	1.5649
			Rank	1	4	5	3	9	8	6	11	7	10	2
F4	100	0	Best	0	0	0	0.072476	61.69	38.012	4.7836	76.288	17.789	69.994	1.04E−06
			Worst	0	3.78E−06	2.1066	0.1004	97.443	53.896	6.0809	87.33	25.396	90.349	0.004369
			Average	0	2.81E−07	0.12536	0.084946	91.549	46.224	5.1918	81.679	20.913	84.429	0.000281
			SD	0	7.38E−07	0.47945	0.007729	9.6427	3.8277	0.28165	2.4978	2.0684	3.9694	0.000794
			Rank	1	2	4	5	11	8	6	9	7	10	3
F5	100	0	Best	98.98	98.04	94.64	98.459	95.665	900.97	2261.6	17,408	213.51	2,966,500	95.041
			Worst	99	98.84	98.149	98.994	556.51	3566.2	4366.6	40,929	1622.1	150,780,000	98.338
			Average	98.987	98.668	95.885	98.811	164.43	1797.5	2914.8	28,836	554.43	38,475,000	97.227
			SD	0.0046505	0.21881	0.91669	0.13719	95.748	632.71	526.66	5621.5	298.55	30,739,000	0.8424
			Rank	5	3	1	4	6	8	10	9	7	11	2
F6	100	0	Best	15.386	18.254	0.93763	15.728	90.646	4.975	121.27	13.078	0.00093	393.22	5.2274
			Worst	24.75	21.197	8.0757	18.04	10,120	94.351	190.19	30.532	0.063177	24,664	8.7994
			Average	20.649	19.751	4.8412	16.919	810.19	16.547	153.35	21.43	0.009123	5010.1	7.4685
			SD	3.0249	0.82578	1.7663	0.58517	1795.2	16.137	16.149	3.6229	0.012269	5103.9	0.95007
			Rank	7	6	2	5	10	4	9	8	1	11	3
F7	100	0	Best	6.523E−08	1.6E−07	7.31E−08	1.2E−07	2.243E−07	6.22E−08	4.06E−07	8.49E−07	1.03E−07	1.888E−07	4.05E−07
			Worst	4.0629E−05	0.000135	4.88E−05	5.69E−05	8.723E−05	5.9E−05	7.49E−05	9.62E−05	6.18E−05	0.0001458	7.38E−05
			Average	1.2803E−06	2.91E−05	1.22E−05	1.79E−05	2.675E−05	2.06E−05	1.79E−05	2.22E−05	1.94E−05	2.638E−05	2.13E−05
			SD	1.2663E−05	3E−05	1.28E−05	1.77E−05	2.63E−05	1.61E−05	2.1E−05	1.95E−05	1.51E−05	3.668E−05	1.97E−05
			Rank	1	10	2	3	9	5	3	7	4	8	6

Table 5 (continued)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWOs
F8	100	− 4189.8	Best	− 16,732	− 11,672	− 22,424	− 13,330	− 24,727	− 26,891	− 25,886	− 19,607	− 27,920	− 8734.6	− 20,141
			Worst	− 10,099	− 7918.4	− 14,136	− 9330.4	− 19,588	− 20,458	− 21,172	− 17,193	− 20,483	− 6449.4	− 6165
			Average	− 14,691	− 9831.6	− 16,282	− 11,500	− 22,249	− 23,936	− 23,164	− 18,225	− 24,002	− 7507.7	− 16,259
			SD	1270.3	1003.9	2244.7	923.65	1307	1511.1	1100.6	484.11	1581.4	542.15	2910.5
			Rank	4	2	6	3	8	10	9	7	11	1	5
F9	100	0	Best	0	0	0	0	283.6	412.01	159.82	630.13	79.598	9.3658	0
			Worst	0	0	24.596	0	629.82	592.24	314.66	721.45	217.9	399.47	5.4266
			Average	0	0	0.88626	0	407.88	505.66	226.99	688.49	129.82	189.98	0.30328
			SD	0	0	4.4927	0	77.086	47.83	28.322	20.836	35.01	102.81	1.1769
			Rank	1	1	3	1	7	8	6	9	4	5	2
F10	100	0	Best	0	0	0	0	10.215	1.9509	2.6682	1.6686	3.9155	3.3783	0
			Worst	0	13.794	1.15E−07	0	19.966	3.1564	3.1778	2.2856	7.7487	20.683	0
			Average	0	0.69234	3.83E−09	0	17.712	2.6163	2.9262	1.9739	5.2556	19.558	0
			SD	0	2.7827	2.1E−08	0	3.0842	0.33796	0.12356	0.16905	0.97947	3.8861	0
			Rank	1	3	2	1	8	5	6	4	7	9	1
F11	100	0	Best	0	0	0	102.9	60.614	0.97535	2.0634	1.1433	0.054574	1.3953	0
			Worst	0	0	0	626.42	303.43	1.2838	2.7183	1.2622	0.23051	145.62	0.014782
			Average	0	0	0	304.32	149.69	1.1274	2.3558	1.1992	0.11727	43.961	0.000908
			SD	0	0	0	140.88	62.202	0.055815	0.13868	0.032296	0.036331	32.802	0.00347
			Rank	1	1	1	9	8	4	6	5	3	7	2
F12	100	0	Best	0.050701	0.66817	0.021836	0.78373	12.64	3.1389	0.19612	25.225	6.5345	24,355,000	0.11424
			Worst	1.1292	0.98226	0.098379	0.88498	30.784	9.512	1.1478	883.24	18.903	271,330,000	0.24632
			Average	0.29802	0.78873	0.063165	0.8306	21.566	5.5966	0.37508	70.444	10.741	123,350,000	0.17789
			SD	0.23335	0.079303	0.020388	0.024106	4.7067	1.3757	0.1907	154.7	3.0408	61,926,000	0.036489
			Rank	3	5	1	6	9	7	4	10	8	11	2
F13	100	0	Best	8.9108	9.4749	1.7654	9.7897	165.22	62.078	5.2607	655.49	126.39	30,603,000	4.9977
			Worst	9.9	9.8951	6.258	10.035	410,060,000	224.82	7.284	8684.3	207.01	608,750,000	6.304
			Average	9.8199	9.7456	3.7413	9.9282	13,669,000	128.74	6.4675	3997.2	164.11	196,570,000	5.5979
			SD	0.20663	0.11467	0.97332	0.06791	74,867,000	34.023	0.46679	2360.2	17.935	127,050,000	0.33688
			Rank	5	4	1	6	10	7	3	9	8	11	2
Mean rank				3.04	4.58	3.27	5.00	9.08	7.00	6.46	8.38	6.00	9.85	3.35
Rank				1	4	2	5	10	8	7	9	6	11	3

Table 6 Comparative results for (F1–F13) set to 1000

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F1	1000	0	Best	0	0	0	1.4394	1,127,400	585,370	13,576	2,433,200	109,810	132,530	0
			Worst	0	0	0	1.7101	1,487,700	815,450	16,943	2,569,200	132,670	660,540	0
			Average	0	0	0	1.5793	1,313,200	709,740	14,526	2,499,200	119,210	387,970	0
			SD	0	0	0	0.069984	95,674	65,388	741.5	30,601	5611.1	142,280	0
			Rank	1	1	1	2	9	7	3	10	6	8	1
F2	1000	0	Best	0	0	123.78	0.00012791	Inf	2724.1	Inf	Inf	819.24	Inf	2.2502E−06
			Worst	0	0	1.6985E+100	0.01359	Inf	2866.2	Inf	Inf	907.97	Inf	0.00010642
			Average	0	0	5.6616E+98	0.0061506	Inf	2794.4	Inf	Inf	861.27	Inf	2.7544E−05
			SD	0	0	3.101E+99	0.0037399	NaN	37.041	NaN	NaN	24.937	NaN	2.8756E−05
			Rank	1	1	7	4	8	6	8	8	5	8	3
F3	1000	0	Best	0	27.192	8048.7	47.113	6,548,200	54,991,000	3,593,300	30,059,000	903,930	13,281,000	285,790
			Worst	0	994,350	404,570	380.36	15,144,000	234,720,000	5,436,500	42,464,000	6,336,500	26,683,000	1,065,800
			Average	0	149,810	144,370	133.96	9,757,500	125,690,000	4,269,100	37,291,000	2,492,400	20,537,000	565,730
			SD	0	244,450	145,120	91.268	2,179,800	37,941,000	423,580	3,135,300	1,367,300	3,047,800	187,650
			Rank	1	4	3	2	10	11	9	6	7	8	5
F4	1000	0	Best	0	3.5482E−06	0	0.17603	94.79	99.314	47.825	Inf	31.978	99.374	64.319
			Worst	0	2.8171	35.461	0.22507	99.748	99.696	54.03	Inf	43.369	99.711	82.215
			Average	0	0.10575	2.545	0.20277	99.125	99.57	50.443	Inf	35.987	99.542	72.052
			SD	0	0.5146	7.971	0.011829	0.97531	0.10051	1.2987	NaN	2.972	0.089836	4.4891
			Rank	1	2	4	3	8	10	7	11	5	9	6
F5	1000	0	Best	998.98	998.45	996.66	999.12	623,400,000	1,375,000,000	1,114,000	14,067,000,000	25,247,000	1,823,400,000	995.94
			Worst	998.99	998.86	11,216	999.73	1,016,800,000	2,164,500,000	2,256,400	15,145,000,000	40,622,000	4,505,900,000	997.14
			Average	998.99	998.77	1338.3	999.31	798,010,000	1,649,000,000	1,425,700	14,721,000,000	31,070,000	3,315,100,000	996.89
			SD	0.0030579	0.072559	1865.7	0.13807	96,970,000	190,360,000	312,920	261,020,000	3,237,100	631,520,000	0.24283
			Rank	3	2	5	4	9	11	7	10	6	8	1
F6	1000	0	Best	249.15	240.11	190.47	237.3	1,129,800	586,270	13,330	2,429,000	107,920	201,250	198.09
			Worst	249.75	245.47	223.09	242.71	1,494,700	803,020	15,539	2,549,300	138,230	583,680	208.33
			Average	249.58	243.65	210.23	240.03	1,317,500	679,630	14,355	2,493,800	118,750	374,780	202.62
			SD	0.169	1.224	9.3985	1.293	91,752	62,740	553.73	34,560	6334	109,680	2.2587
			Rank	5	4	2	3	11	10	6	9	7	8	1
F7	1000	0	Best	2.724E−08	3.1537E−07	4.5776E−07	5.3879E−07	5.0278E−07	1.5177E−06	7.64E−07	1.3506E−06	3.1999E−07	1.3786E−07	3.8969E−07
			Worst	5.374E−05	0.00010104	0.000043091	6.2046E−05	9.5559E−05	7.4232E−05	4.93E−05	0.000096299	0.00012816	0.000097588	6.7353E−05
			Average	1.397E−05	2.8122E−05	0.000011294	2.2603E−05	2.8489E−05	2.3481E−05	1.76E−05	0.000029333	2.9303E−05	0.000025139	1.8128E−05
			SD	1.408E−05	2.8384E−05	0.000011204	1.8272E−05	2.8736E−05	1.9827E−05	1.42E−05	0.000027963	3.5551E−05	0.000027001	2.0531E−05
			Rank	1	6	9	5	7	4	2	11	8	10	3

Table 6 (continued)

Function	Dim	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F8	1000	−418,980	Best	− 62,863	− 34,763	− 77,097	− 40,754	− 85,956	− 111,850	− 178,590	− 59,057	− 143,010	− 29,147	− 134,800
			Worst	− 29,336	− 27,783	− 44,762	− 32,017	− 70,074	− 31,509	− 147,360	− 54,303	− 120,550	− 20,308	− 20,919
			Average	− 56,492	− 30,601	− 52,011	− 36,681	− 80,079	− 68,269	− 163,140	− 56,638	− 131,160	− 23,805	− 103,200
			SD	5949	2014.5	8108	2375.5	3670.8	18,530	8472.6	1088.5	6753.3	2221.5	23,589
			Rank	5	2	4	3	8	7	11	6	10	1	9
F9	1000	0	Best	0	0	0	6.0831E−06	6144.2	9350.6	6065.8	15,799	5267.7	516.5	0
			Worst	0	0.0078655	2.0429	6.3209E−05	7128	11,407	6793.7	16,119	5792.1	4554.2	43.111
			Average	0	0.00026221	0.13598	4.1984E−05	6667	10,121	6318.6	15,965	5592	2097.1	12.564
			SD	0	0.001436	0.51749	1.1939E−05	281.54	443.23	207.41	84.753	127.21	961.67	9.542
			Rank	1	3	4	2	8	9	10	11	7	6	5
F10	1000	0	Best	0	0	0	0.0082227	19.962	19.822	5.9931	20.922	12.435	9.7864	5.1463E−07
			Worst	0	0	5.9865E−06	0.0093529	20.401	20.18	6.9134	20.945	13.157	20.83	9.1297E−07
			Average	0	0	3.1673E−07	0.0088694	20.032	20.057	6.1699	20.933	12.781	19.054	7.4656E−07
			SD	0	0	1.2447E−06	0.0002792	0.12624	0.07611	0.17303	0.0069186	0.17468	3.7009	1.0843E−07
			Rank	1	1	2	4	8	9	5	10	6	7	3
F11	1000	0	Best	0	0	0	27,584	13,604	4966.5	122.4	21,837	964.12	2200.6	0
			Worst	0	0	1.0562E−06	28,637	18,616	7388.8	164.05	23,173	1142	5607.9	0.049039
			Average	0	0	4.0559E−08	28,177	15,945	6002.1	134.21	22,560	1055.2	3406.4	0.0023362
			SD	0	0	1.9404E−07	305.02	980.29	667.2	8.7527	309.89	50.353	925.88	0.0096206
			Rank	1	1	2	10	8	7	4	9	5	6	3
F12	1000	0	Best	1.1798	1.1112	0.50586	1.078	1,110,500,000	1,850,900,000	55,601	35,815,000,000	15,299	5,355,900,000	0.80334
			Worst	1.1923	1.1493	0.85382	1.1079	2,107,600,000	3,088,200,000	197,350	40,836,000,000	226,520	14,433,000,000	0.87024
			Average	1.1891	1.1291	0.76934	1.0923	1,495,500,000	2,464,300,000	96,418	38,616,000,000	81,049	9,715,500,000	0.84123
			SD	0.0028533	0.0086047	0.071531	0.0065697	248,840,000	324,690,000	34,066	1,025,300,000	61,344	2,201,900,000	0.017605
			Rank	5	4	1	3	10	11	6	8	7	9	2
F13	1000	0	Best	99.9	99.78	98.316	100.37	3,442,500,000	4,180,900,000	444,710	63,525,000,000	7,042,100	8,826,000,000	93.603
			Worst	100	99.982	99.224	100.53	5,782,200,000	7,328,600,000	1,284,200	70,496,000,000	21,769,000	20,948,000,000	96.582
			Average	99.904	99.878	98.892	100.46	4,487,200,000	5,637,900,000	725,760	67,879,000,000	14,752,000	15,157,000,000	95.274
			SD	0.018235	0.042193	0.20082	0.042065	682,600,000	674,620,000	219,580	1,489,900,000	4,273,300	3,528,300,000	0.64137
			Rank	4	3	2	5	11	10	6	8	7	9	1
Mean rank				2.88	3.79	3.21	4.75	8.42	8.83	5.58	10.17	6.42	8.67	3.29
Rank				1	4	2	5	8	10	6	11	7	9	3

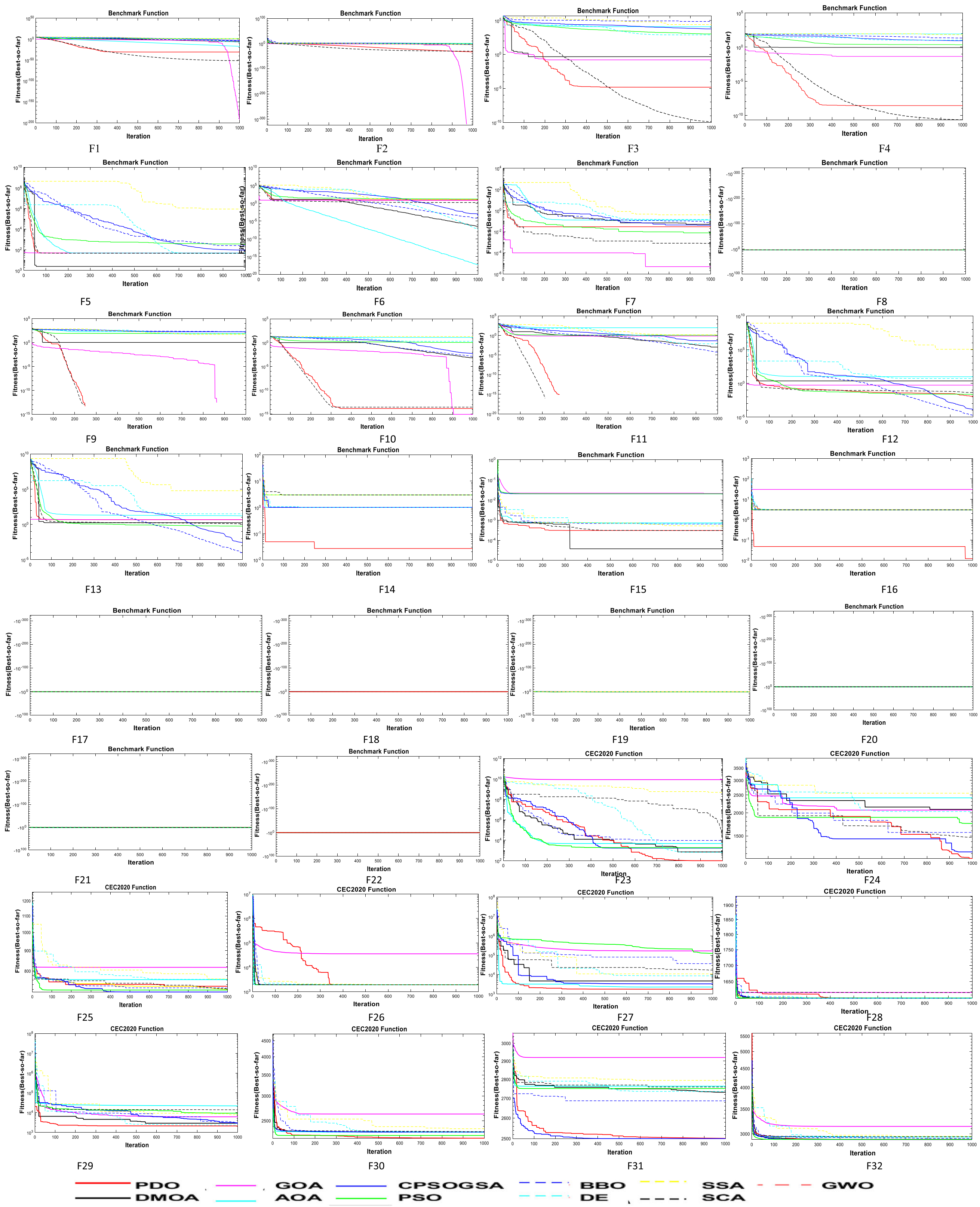


Fig. 7 Convergence behaviour results for both classical and CEC 2020 benchmark function

Table 7 Comparative results for CEC 2020

Function	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F23	100	Best	100.06	1.12E+07	104.21	1.81E+09	100.69	100.47	100.96	142.99	107.53	2.44E+08	5155.8
		Worst	101.58	4.10E+09	11,449	1.29E+10	12,694	5621.7	5494.9	7359.8	12,243	1.29E+09	5.07E+08
		Average	100.46	7.29E+08	2859.6	6.54E+09	4722.1	1692.5	876.9	2795	3236	7.06E+08	1.89E+07
		SD	0.35113	1.21E+09	2965.3	2.57E+09	4289.2	1682.6	1302.9	2097.7	2872.5	2.98E+08	9.23E+07
		Rank	1	9	5	11	3	2	4	7	6	10	8
F24	1100	Best	1130.8	1453.2	1126.9	1359.4	1594.2	1118.5	1115.2	1160.7	1138.5	1552.1	1112.4
		Worst	1375.1	2666.7	2567.9	2707.7	2832	1823.1	2263.4	1724.3	2333.4	2600.6	2062.1
		Average	1241.4	2194.6	1709.7	2021.1	2034.2	1453.2	1590.8	1552.1	1766.4	2269.5	1513.2
		SD	63.833	288.25	305.26	288.74	298.97	171.22	281.95	131.95	262.51	275.68	208.43
		Rank	1	10	6	8	9	2	5	4	7	11	3
F25	700	Best	714.03	747.27	720.96	769.12	720.77	711.97	714.7	717.27	716.57	753.67	713.28
		Worst	724.97	835.07	799.36	834.54	883.67	729.72	730.06	726.09	772.57	799.94	754.41
		Average	719.69	778.54	745.26	797.74	761.4	720.38	720.67	722.33	732.7	774.21	733.63
		SD	2.963	19.441	22.3	15.613	31.878	3.7772	4.1885	2.1465	12.868	9.7638	11.536
		Rank	1	10	7	11	8	2	3	4	5	9	6
F26	1900	Best	1900.6	1903.1	1900.8	2571.8	1900.1	1900.5	1900.5	1901	1900.4	1909.1	1900.7
		Worst	1901.8	27,556	1916	2.00E+05	1906.5	1901.5	1905.1	1901.9	1903.4	2048.7	1903.5
		Average	1901.2	3264.3	1905.3	52,046	1901.8	1901	1901.5	1901.6	1901.5	1926.5	1902
		SD	0.3165	5087.8	2.9321	57,296	1.2935	0.28017	0.93587	0.22955	0.73147	27.946	0.80342
		Rank	1	8	6	9	4	1	2	3	3	7	5
F27	1700	Best	1720.1	3674.5	1868.4	3679.4	2268.2	1957.2	2682.9	3655.3	3015	6282	2695.1
		Worst	1797.9	5.71E+05	5.67E+05	3.10E+05	1.22E+05	18,436	2.18E+05	79,843	20,479	84,903	3.47E+05
		Average	1751.3	69,217	84,962	1.55E+05	19,498	6326.5	39,375	24,258	6712.5	22,104	17,781
		SD	16.644	1.63E+05	1.69E+05	70,051	30,567	4401.8	57,716	19,590	3874.4	18,350	62,170
		Rank	1	9	10	11	5	2	8	7	3	6	4
F28	1600	Best	1600.5	1600.9	1600.5	1601.1	1600.5	1600.5	1600.5	1600.5	1600.5	1600.8	1600.8
		Worst	1600.5	1659.5	1619.5	1628.8	1727.1	1659.5	1618	1600.8	1601.6	1602.1	1617.5
		Average	1600.5	1606.2	1603.7	1615.1	1626.1	1617.2	1603.2	1600.6	1601	1601.5	1602.9
		SD	0.001129	12.062	5.708	8.062	34.999	20.581	5.8731	0.073957	0.29051	0.28056	5.1751
		Rank	1	8	7	9	11	10	6	2	3	4	5
F29	2100	Best	2101.5	4674	2102.1	3334.3	2891.9	2101.8	2121.4	2208.8	2438.6	4020	2450.9
		Worst	2122.5	27,880	26,801	10,767	23,780	5640.2	24,664	5785	21,675	19,690	19,648
		Average	2108	11,226	9730.3	6361.5	9699.7	2741.6	9666.2	3097.4	7310	9433.6	9618
		SD	4.5969	6225.9	7716.8	2263.6	7186.9	746.11	7064.8	1025.2	5184.9	4470.4	5057.1
		Rank	1	11	10	4	9	2	8	3	7	5	6

Table 7 (continued)

Function	Global	Value	PDO	DMOA	GOA	AOA	CPSOGSA	PSO	BBO	DE	SSA	SCA	GWO
F30	2200	Best	2200	2238.5	2238.3	2456.4	2301.1	2301.1	2222.3	2236.9	2222.9	2332.3	2300.9
		Worst	2304.4	2544.7	2321.4	3388.6	4219.3	2305.1	2304.2	2301.5	2305.7	2424.8	2327.9
		Average	2278.3	2362.6	2302.8	2770.9	2428.8	2302.4	2299.4	2295.9	2295.9	2368.9	2307.5
		SD	43.878	71.069	21.088	237.13	424.81	1.0982	14.607	15.238	23.22	22.146	6.9467
		Rank	1	7	5	9	10	4	3	2	2	8	6
F31	2400	Best	2500	2521.9	2500	2679.8	2500	2500	2500	2599	2722.3	2552.3	2501.3
		Worst	2739.7	2830.3	2830.6	2975.7	2804.9	2753.9	2773.4	2753.2	2784.5	2794.4	2777.8
		Average	2559.5	2733.4	2781.8	2823.7	2743.4	2717.7	2722.1	2732	2745.1	2772.6	2736.7
		SD	83.741	97.457	56.986	66.405	83.431	74.071	75.875	36.434	10.568	42.3	47.034
		Rank	1	5	10	11	7	2	3	4	8	9	6
F32	2500	Best	2602.7	2901	2600.2	2908.4	2897.7	2898.1	2897.7	2900	2897.8	2926	2902.6
		Worst	2897.8	3055.1	3024.3	3433.7	3024.2	2946	2948.9	2946.2	2951.3	3041.5	2949.5
		Average	2887.9	2955.5	2938.1	3121.4	2931.6	2924.7	2930.5	2918.1	2924.2	2959.9	2936.8
		SD	53.863	38.61	72.682	113.74	30.553	23.104	23.126	16.173	24.521	23.205	14.062
		Rank	1	9	8	11	7	4	5	2	3	10	6
Mean rank			1.10	9.00	7.70	9.80	7.80	3.30	4.75	3.75	4.60	8.20	6.00
Rank			1	10	7	11	8	2	5	3	4	9	6

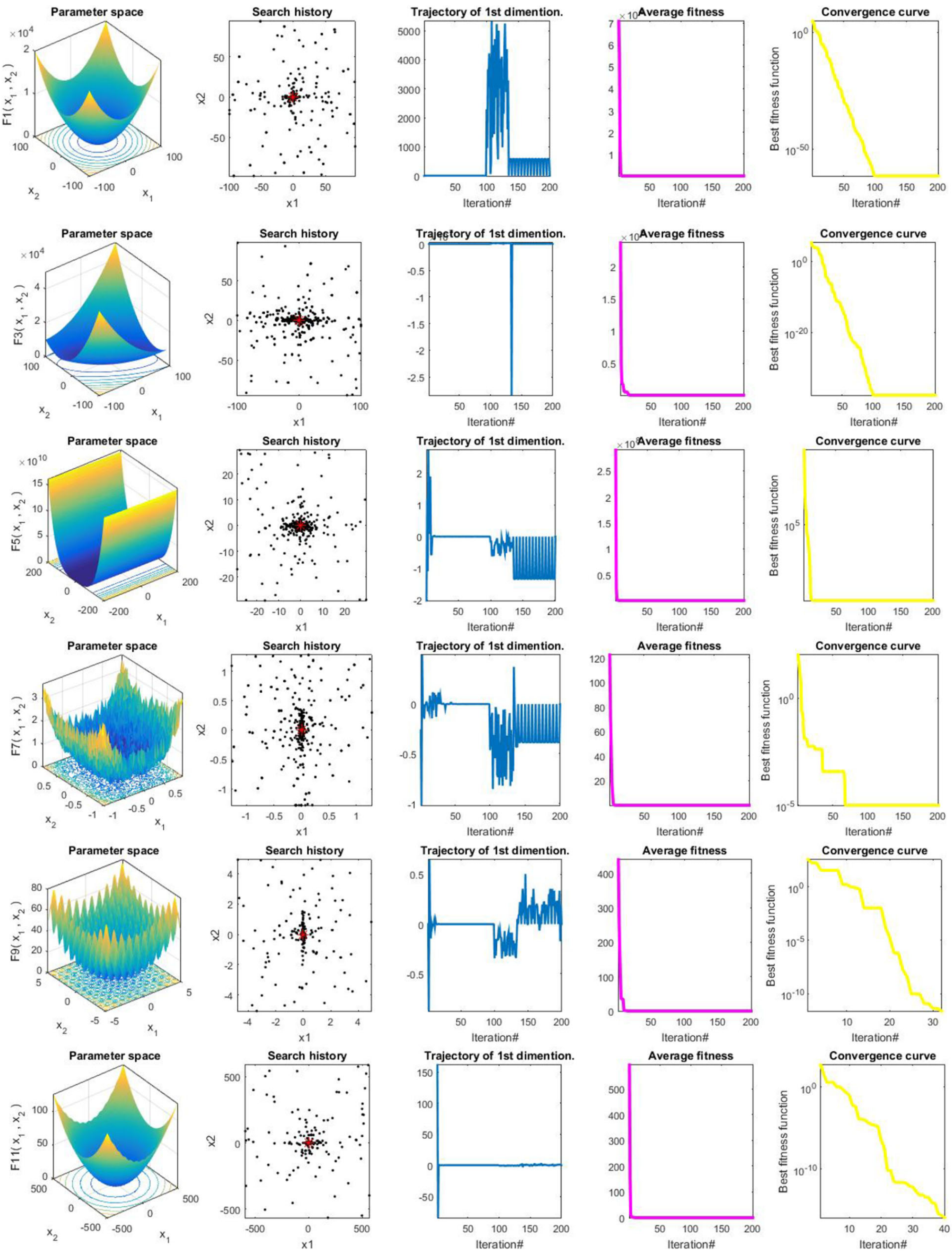


Fig. 8 Qualitative results metrics, including 3D parameter space, search history, the trajectory of the first dimension, average fitness, and convergence curve

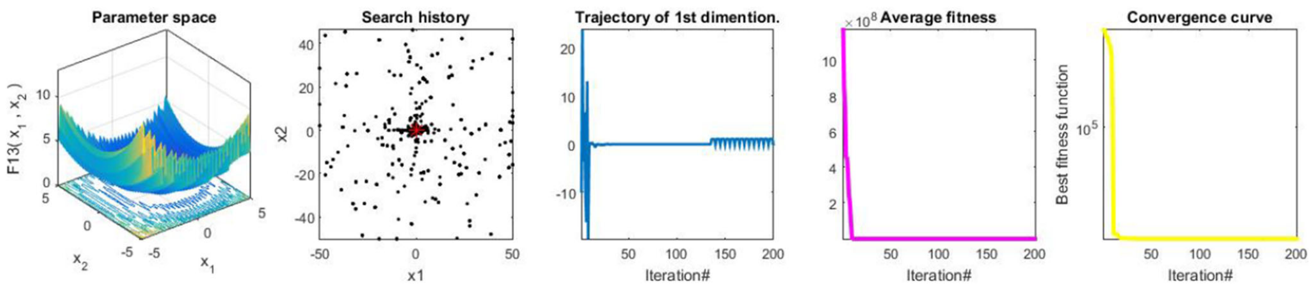


Fig. 8 continued

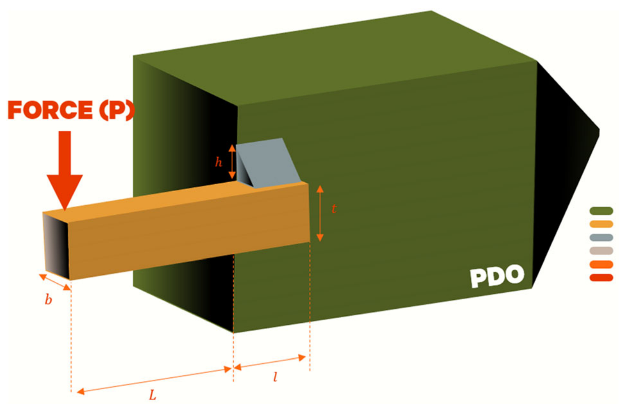


Fig. 9 Schematic diagram of the WBD

vessel, as shown in Fig. 11. Four design variables control the optimization process of PVD, namely: the inner radius (R), the thickness of the head (T_h), the length of the cylindrical section of the vessel (L), and the thickness of the shell (T_s). The mathematical model of the PVD is shown in Eq. 16 [70].

Given

$$l = [T_s T_h R L] = [x_1, x_2, x_3, x_4],$$

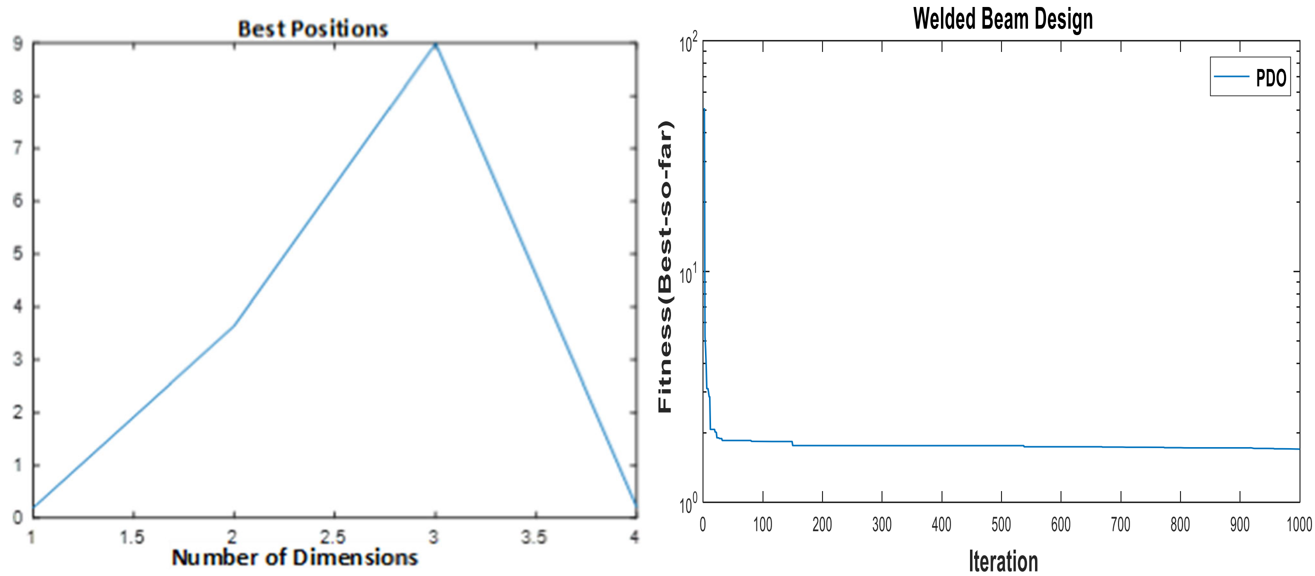
$$Minf\left(\overrightarrow{l}\right) = 0.6224x_1x_3x_4 1.781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

$$\begin{aligned} g_1\left(\overrightarrow{x}\right) &= -x_1 + 0.0193x_3 \le 0, g_2\left(\overrightarrow{x}\right) \\ &= -x_3 + 0.00954x_3 \le 0, g_3\left(\overrightarrow{x}\right) \\ &= -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0, g_4\left(\overrightarrow{x}\right) \\ &= x_4 - 240 \le 0. \end{aligned} \tag{16}$$

The design variables defined in the interval are given below:

Table 8 Comparative results for the WBD

Algorithm	x_1	x_2	x_3	x_4	Best	Worst	Average	SD	FEs
PDO	0.196302698	3.493203912	9.0855552	0.207038986	**1.6882**	1.7286	**1.6970**	0.000147	9,000
DMOA [58]	0.205234326	3.26225324	9.037040824	0.205730354	1.7115	1.7685	1.7287	0.014025	20,000
AOA [58]	0.247498553	2.536812819	10	0.250284392	1.8417	2.7377	2.2166	0.26738	30,000
CPSOGSA [58]	0.20573	3.470489	9.036624	0.20573	1.7009	2.2115	1.8406	0.13769	40,000
PSO [27]	0.2057	3.4705	9.0366	0.2057	1.09E+14	1.09E+14	1.09E+14	0.047676	60,000
BBO [27]	0.205563	3.484293	9.03662	0.236204	1.09E+14	1.09E+14	1.09E+14	0.16021	100,000
DE [58]	0.205396	3.4702	9.036624	0.205806	1.09E+14	1.09E+14	1.09E+14	0.047676	30,000
SSA [58]	0.205234326	3.4704887	9.037040824	0.205730354	1.6965	2.0123	1.7788	0.094081	45,000
SCA [58]	0.205563	3.4705	9.03662	0.205811	1.7223	1.878	1.8156	0.036014	50,000
GWO [58]	0.205396	3.470489	9.03495	0.205796	1.6958	1.7016	1.6971	0.0014794	55,000
ICA [66]	0.205799	3.469634	9.03495	0.205806	1.725135	2.237755	1.79433	1.10E−01	50,000
WCA [67]	0.205728	3.470522	9.03662	0.205729	1.724856	1.744697	1.726427	4.29E−03	46,500
ABC [67]	0.20573	3.470489	9.036624	0.20573	1.724852	NA	1.741913	3.10E−02	30,000
EO [68]	0.2057	3.4705	9.0366	0.2057	1.724853	1.736725	1.726482	3.26E−03	15,000
TEO [39]	0.205681	3.472305	9.035133	0.205796	1.725284	1.931161	1.76804	5.82E−02	NA
WSA [39]	0.20573	3.470489	9.036624	0.20573	1.724852	1.725068	1.724908	4.15E−05	50,000
SHO [66]	0.205563	3.474846	9.035799	0.205811	1.725661	1.726064	1.725828	2.87E−04	NA
WOA [66]	0.205396	3.484293	9.037426	0.206276	1.730499	NA	1.732	0.0226	9900
SNS [69]	0.2057296	3.4704887	9.0366239	0.2057296	1.724852	1.725051	1.72488	5.18E−05	9000



(a) Best positions (b) Convergence curve

Fig. 10 The convergence curve for the WBD

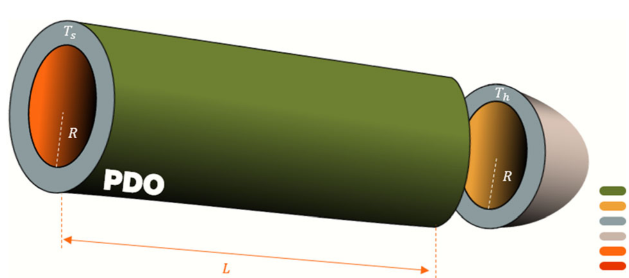


Fig. 11 The schematic illustration of the PVD

$0 \leq x_1 \leq 99, 0 \leq x_2 \leq 99, 10 \leq x_3 \leq 200, 10 \leq x_4 200$ [70]

The results of PDO and the other competitive algorithms are given in Table 9. It can be observed that the PDO returns the least value for the objective function, and it also has a mean value second to DMOA. However, the DMOA got trapped at the best value and did not explore other areas, making the PDO more balanced and stable. From Fig. 12, the PDO converged to the optimal solution early at the 100th iteration. The figure reflects that the PDO steadily moved between the exploration and exploitation phases.

4.2.3 The compression spring design problem (CSD)

The compression spring design (CSD) is an optimization problem that aims to minimize the weight of a tension/compression spring given the values of three parameters: the wire diameter (d), number of active coils (P), and mean coil diameter (D), respectively. Figure 13 shows the schematic diagram of the CSD, and Eq. 17 shows the model of the objective function of CSD [72].

Given that

$$l = [x_1 x_2 x_3] = [dDP]$$

$$Minf(\overrightarrow{x}) = (x_3 + 2)x_2 x_1^2 \tag{17}$$

subject to

$$g_1(\overrightarrow{x}) = 1 - \frac{x_2^3 x_3}{717851^4} \leq 0,$$

$$g_2(\overrightarrow{x}) = \frac{4x_2^2 - x_1 x_2}{12566(x_3 x_1^3 - x_1^4)} + 1/5108{x_1}^2 \leq 0,$$

$$g_3(\overrightarrow{x}) = 1 - \frac{140.45x_1}{x_2^2 x_3} \leq 0,$$

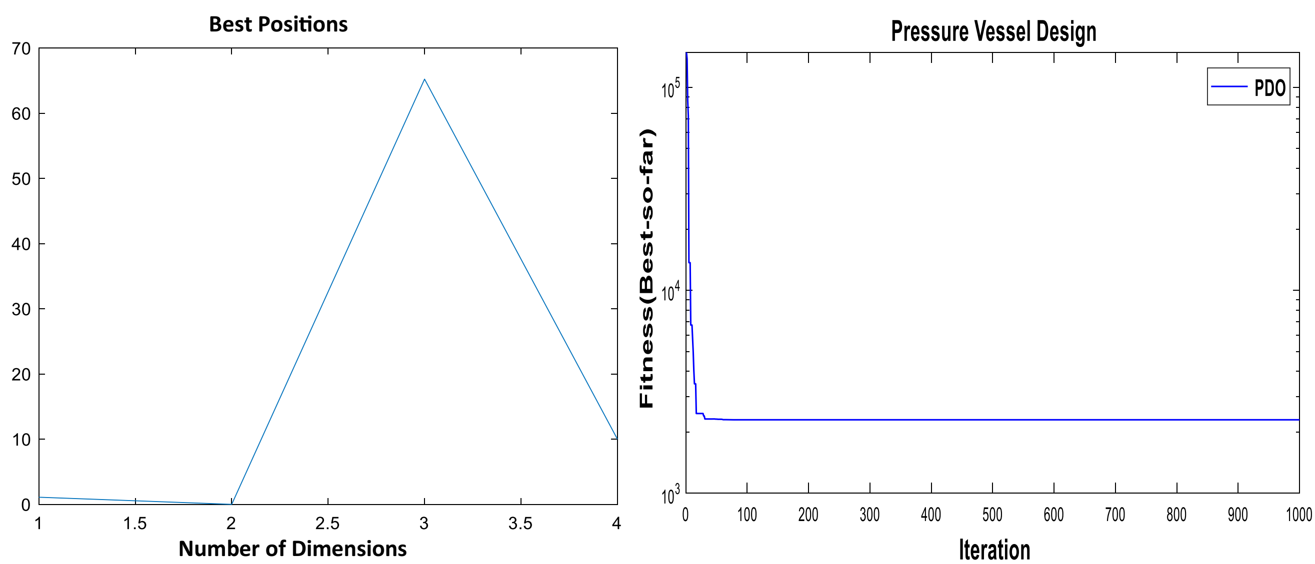
$$g_4(\overrightarrow{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

The intervals for the design variables are: $0.05 \leq x_1 \leq 2.00, 0.25 \leq x_2 \leq 1.30,\ 2.00 \leq x_3 \leq 15.0$ [72].

The results for PDO and other competitive algorithms used in this study are given in Table 10. All the algorithms were able to find the optimal weight; the advantage of PDO is that it returned the least value for both the 'mean' and 'best' performance indicators. Furthermore, Fig. 14 shows the behaviour of PDO while solving the CSD problem. The PDO converged early during the optimization process and remained steadily at the optimal solution till the end of the iterations.

Table 9 Comparative results for PVD

Algorithm	x_1	x_2	x_3	x_4	Best	Worst	Average	SD	FEs
PDO	0.793771	0.253932	49.05817	105.9976	**4527.2**	4545.7	4528.5	1.16E−02	5,000
DMOA [58]	0.446319	0.230237	40.33173	200	4527.3	4527.3	**4527.3**	2.96E−11	7,000
AOA [58]	0.579194	0.241237	40.37747	200	5137.9	7326	5984.3	793.96	60,000
CPSOGSA [58]	0.462095	0.240237	40.35339	199.5304	4527.3	5578.9	4982.3	462.54	100,000
PSO [27]	10	10	53.67151	71.64606	2.04E+05	2.04E+05	2.04E+05	1.0906	40,000
BBO [27]	10	10	56.07885	56.40497	2.04E+05	2.08E+05	2.05E+05	1055	25,000
DE [58]	10	10	53.47238	72.98132	2.04E+05	2.04E+05	2.04E+05	0.12325	150,000
SSA [58]	0.69441	0.253932	52.93147	76.66704	4529.1	5567.2	4792.3	305.62	100,000
SCA [58]	0.461506	0.240237	40.32483	200	5112.3	7544	7055.7	989.75	30,000
GWO [27]	0.461984	0.243487	40.31999	200	4527.4	7535.1	4943.7	832.07	40,000
SAP [69]	0.8125	0.4375	40.3239	200	6288.745	6308.15	6293.843	7.41E+00	30,000
HPSO [67]	0.8125	0.4375	42.0984	176.6366	6059.7143	6288.677	6099.9323	8.62E+01	81,000
CDE [68]	0.8125	0.4375	42.0984	176.6376	6371.046	6059.734	6085.23	4.30E+01	204,800
CPSO [71]	0.8125	0.4375	42.0913	176.7465	6061.0777	6363.8041	6147.1332	8.65E+01	200,000
QPSO [71]	0.8125	0.4375	42.0984	176.6374	6059.7209	8017.2816	6839.9326	4.79E+02	8000
G-QPSO [62]	0.8125	0.4375	42.0984	176.6372	6059.7208	7544.4925	6440.3786	4.48E+02	8000
ABC [67]	0.8125	0.4375	42.09845	176.6366	6059.71474	NA	6245.30814	2.05E+02	30,000
CS [67]	0.8125	0.4375	42.09845	176.6366	6059.71434	6495.347	6447.736	5.03E+02	15,000
WOA [63]	0.8125	0.4375	42.0982699	176.638998	6059.741	NA	6068.05	6.57E+01	6300
APSO [70]	0.8125	0.4375	42.0984	176.6374	6059.7242	7544.49272	6470.71568	3.27E+02	200,000
EO [68]	0.8125	0.4375	42.09845	176.6366	6059.7143	7544.4925	6668.114	5.66E+02	15,000
CGO [67]	0.8125	0.4345	42.089181	176.758731	6247.67282	6330.95869	6250.95735	1.07E+01	100,000
SNS [69]	0.8125	0.4375	42.09845	176.6366	6059.71434	6410.08689	6097.10029	9.28E+01	6000



(a) Best position (b) Convergence curve

Fig. 12 Convergence curve for PVD

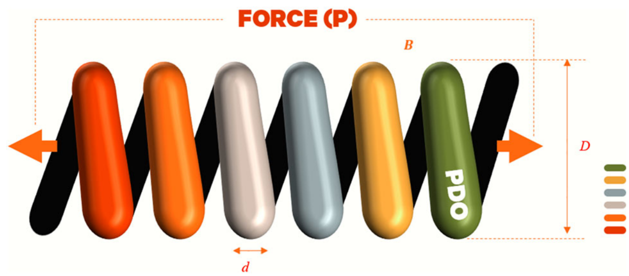


Fig. 13 Schematic diagram of the CSD

4.2.4 The speed reducer design problem (SRD)

As shown in Fig. 15, the speed reducer problem depicts a gearbox that sits between the propeller and engine of an aeroplane; the goal is for the propellers to rotate to guarantee effective speed. In optimization, the SRD minimizes the weight of the SRD subject to the following constraints: the bending stress of the gear teeth, surface stress, transverse deflections of the shafts, and stresses [74]. To solve this problem, seven (7) design variables are considered, namely: x_1 is the face width (b), x_2 is the module of teeth (m), x_3 stands for the number of teeth in the pinion (z), x_4 represents the length of the first shaft between bearings (l_1), (l_2) the length of the second shaft between the bearings is x_5, and the diameter of first (d_1) and second shafts (d_2) is denoted by x_6 and x_7, respectively. The mathematical model of the SRD is given in Eq. 18.

Given

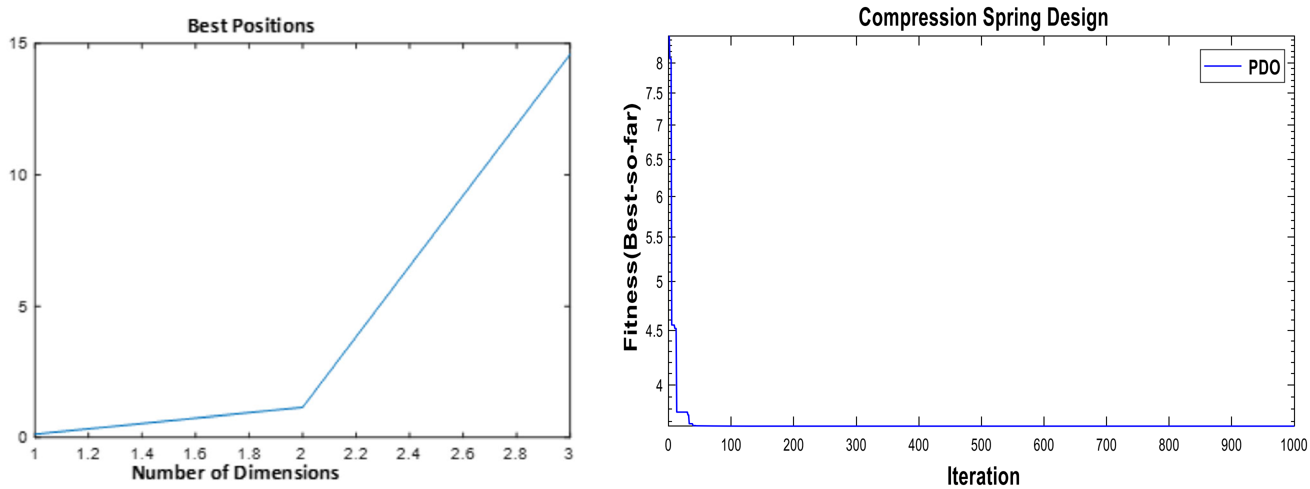
$$x = [x_1, x_2, x_3, x_4, x_5, x_6, x_7] = [b, m, z, l_1, l_2, d_1, d_2]$$

$$\begin{aligned} \min f(x) = {} & 0.7854 x_1 x_2^2 \left(3.3333 x_3^2 + 14.9334 x_3 - 43.0934\right) \\ & - 1.508 x_1 \left(x_6^2 + x_7^2\right) + 7.4777 \left(x_6^3 + x_7^3\right) \\ & + 0.7854 \left(x_4 x_6^2 + x_5 x_7^2\right) \end{aligned} \tag{18}$$

Subject to

Table 10 Comparative results for CSD

Algorithm	d	D	N	Best	Worst	Average	SD	FEs
PDO	0.051749	0.358179	11.203763	**0.01266051**	0.0127	**0.0127**	1.28E−02	3000
DMOA [58]	0.139149732	1.3	11.89243067	0.012665	0.0127763	0.012703	1.34E−03	10,000
AOA [58]	0.148316747	1.3	15	0.0126652	0.0127702	0.012709	1.58E−05	9,000
CPSOGSA [58]	0.136308631	1.220636766	13.22990784	0.01266523	0.012769	0.0127196	2.70E−05	5,000
PSO [58]	2	1.2	2	0.012666	0.012766	0.01273	3.00E−04	8,000
BBO [27]	2.000080395	1.2	2	0.012669	0.01276523	0.012746	3.79E−05	20,000
DE [27]	2	1.2	2	0.0126702	0.0127652	0.012762	5.20E−05	NA
SSA [27]	1	1.2	3	0.0126747	0.012765	0.012760	6.28E−04	10,000
SCA [58]	0.137850353	1.26129434	12.53614501	0.0126763	0.0127659	0.01267653	6.85E−04	NA
GWO [58]	0.138880781	1.3	11.97370703	0.0127	0.01276051	0.012754	8.06E−05	NA
CPSO [68]	0.051728	0.357644	11.244543	0.0126747	0.012924	0.01273	5.20E−05	200,000
HPSO [73]	NA	NA	NA	0.0126652	0.0127191	0.0127072	1.58E−05	81,000
CDE [73]	0.051689	0.356718	11.288968	0.0126702	0.01279	0.012703	2.70E−05	204,800
QPSO [40]	0.0518	0.359	11.279	0.012669	0.018127	0.013854	1.34E−03	20,000
G-QPSO [62]	NA	NA	NA	0.012666	0.015869	0.012996	6.28E−04	20,000
WCA [62]	0.05168	0.3565	11.3004	0.012665	0.012952	0.012746	8.06E−05	11,750
ABC [40]	0.051749	0.358179	11.203763	0.012665	NA	0.012709	1.28E−02	30,000
APSO [72]	0.052588	0.378343	10.138862	0.0127	0.014937	0.013297	6.85E−04	120,000
IAPSO [72]	0.051685	0.356629	11.294175	0.01266523	0.01782864	0.01367653	1.57E−03	20,000
WOA [66]	0.0512	0.3452	12.004	0.0126763	NA	0.0127	3.00E−04	4410
MCEO [73]	0.051994	0.364109	10.868421	0.01266051	0.01350901	0.0127196	3.79E−05	2000
EO [68]	0.051207	0.345215	12.004032	0.012666	0.013997	0.013017	3.91E−04	15,000
SNS [69]	0.051587	0.354268	11.434058	0.01266525	0.01276587	0.01268472	2.38549E−05	9000



(a) Best position **(b)** Convergence curve

Fig. 14 Convergence behaviour for CSD

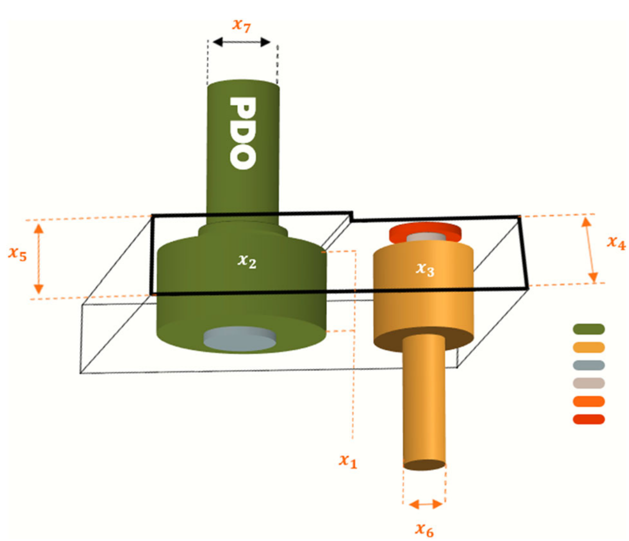


Fig. 15 Schematic diagram of the SRD

$$g_1(x) = \frac{27}{x_1 x_2^2 x_3} - 1 \le 0, g_2(x) = \frac{397.5}{x_1 x_2^2 x_3^2} - 1 \le 0,$$

$$g_3(x) = \frac{1.93 x_4^2}{x_2 x_6^4 x_3} - 1 \le 0, g_4(x) = \frac{1.93 x_5^2}{x_2 x_7^4 x_3} - 1 \le 0,$$

$$g_5(x) = \frac{\sqrt{\left(\frac{745 x_4}{x_2 x_3}\right)^2 + 16 \times 10^6}}{110 x_6^3} - 1 \le 0,$$

$$g_6(x) = \frac{\sqrt{\left(\frac{745 x_5}{x_2 x_3}\right)^2 + 157.5 \times 10^6}}{85 x_7^3} - 1 \le 0,$$

$$g_7(x) = \frac{x_2 x_3}{40} - 1 \le 0, g_8(x) = \frac{5 x_2}{x_1} - 1 \le 0,$$

$$g_9(x) = \frac{x_1}{12 x_2} - 1 \le 0, g_{10}(x) = \frac{1.5 x_6 + 1.9}{x_4} - 1 \le 0,$$

$$g_{11}(x) = \frac{1.1 x_7 + 1.9}{x_5} - 1 \le 0$$

The interval is given as follows: $2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8, 17 \le x_3 \le 28, 7.3 \le x_4 \le 8.3, 7.3 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9, 5.0 \le x_7 \le 5.5$ [74].

The comparative results of PDO and other metaheuristic algorithms available in the literature are given in Table 11. It can be noticed that the PDO returned a value for the objective function second to the CPSOGSA. However, the CPSOGSA never explored any other region as it got trapped at the best value. The PDO outperformed the rest of the other competing algorithms. The qualitative results (best position and convergence curve) of the PDO are shown in Fig. 16. PDO is observed to converge steadily towards the optimal solution beginning with the exploration phase and then changing to exploitation to obtain the solution in the feasible region.

4.2.5 The three-bar truss design problem (3-BTD)

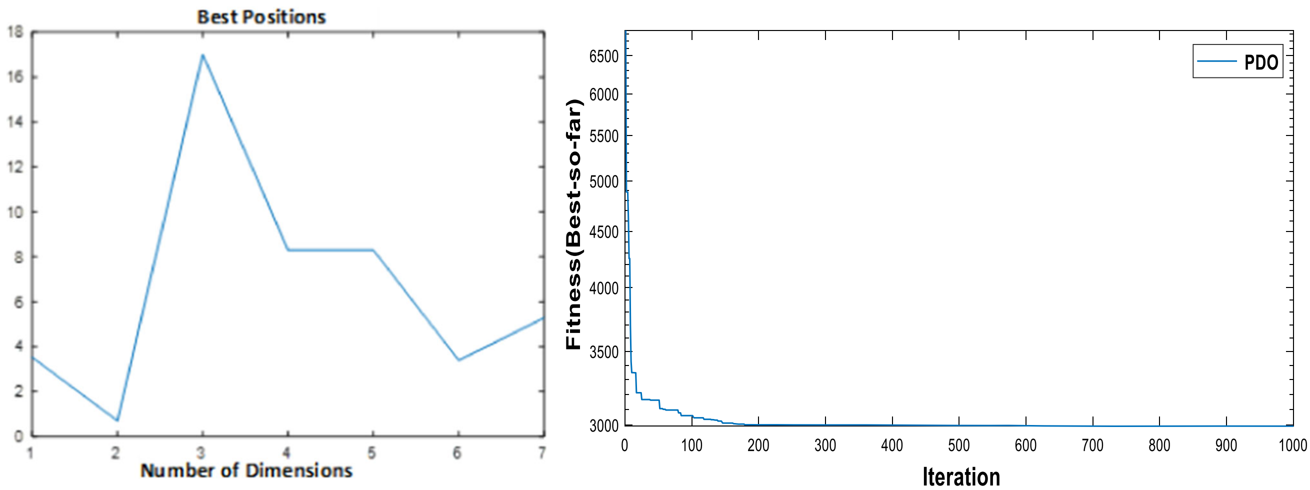
The objective of the three-bar truss problem shown in Fig. 17 is to minimize three-bar structure weight subject to supporting a total load P acting vertically downwards. The static volume of the loaded three-bar truss is subject to each bar's stress (σ) constraints. Two design variables for the 3-BTD are, namely, the cross-sectional areas, $A_1 (= x_1) and A_2 (= x_2)$ as illustrated in Fig. 17. The 3-BTD is mathematically modelled as in Eq. 19 [75].

$$\min f(X) = \left(2\sqrt{2} x_1 + x_2\right) \times l \tag{19}$$

Subject to

Table 11 Comparative results for SRD

Algorithm	x1	x2	x3	x4	x5	x6	x7	Best	Worst	Average	SD	FEs
PDO	3.497777468	0.7	17.00002761	7.300100314	7.800675175	3.351095015	5.296455378	2993.7	3010	**2999.5**	3.888	2,700
DMOA [58]	3.497599093	0.7	17	7.3	7.713534977	3.350055806	5.285631197	3010.4	3097.8	3046.7	19.919	6,000
AOA [58]	3.5	0.7	17	7.3	7.71532	3.35021	5.28665	3058.7	3221.9	3146.5	48.103	20,000
CPSOGSA [58]	3.5015	0.7	17	7.605	7.8181	3.352	5.2875	**2993.6**	2993.6	2993.6	5.00E−13	50,000
PSO [58]	3.5015	0.7	17	7.605	7.8181	3.352	5.2875	1.32E+07	1.32E+07	1.32E+07	4.17E−09	10,000
BBO [27]	3.5	0.7	17	7.3	7.8	3.35022	5.28668	1.32E+07	1.32E+07	1.32E+07	3.79E−09	20,00
DE [27]	3.5	0.7	17	7.3	7.71532	3.35021	5.28665	1.32E+07	1.32E+07	1.32E+07	4.09E−09	22,000
SSA [27]	3.50131	0.7	18	8.12781	8.04212	3.35245	5.28708	2994.7	3096.5	3017	22.261	NA
SCA [58]	3.497599093	0.7	17	7.3	7.713534977	3.350055806	5.285631197	3028.9	3154.6	3087.8	31.309	NA
GWO [58]	3.5	0.7	17	7.3	7.71532	3.35021	5.28665	2995.3	3008.5	3000.5	3.5931	NA
CS [67]	3.5015	0.7	17	7.605	7.8181	3.352	5.2875	3000.981	3009	3007.1997	4.96E+00	250,000
ABC [67]	3.5	0.7	17	7.3	7.8	3.35022	5.28668	2997.05841	NA	2997.058412	0.00E+00	30,000
WCA [62]	3.5	0.7	17	7.3	7.71532	3.35021	5.28665	2994.47107	2994.50558	2994.474392	7.40E−03	15,150
APSO [62]	3.50131	0.7	18	8.12781	8.04212	3.35245	5.28708	3187.63049	4443.01764	3822.640624	3.66E+02	30,000
SHO [72]	3.50159	0.7	17	7.3	7.8	3.35127	5.28874	2998.5507	3003.889	2999.64	1.93E+00	NA
WOA [72]	NA	NA	NA	NA	NA	NA	NA	2996.60434	3233.59812	3042.915023	4.08E+01	NA
CSS [72]	NA	NA	NA	NA	NA	NA	NA	2996.49248	3106.21645	3005.658912	4.86E+00	NA
CGO [69]	NA	NA	NA	NA	NA	NA	NA	2994.44365	2995.50493	2994.465397	0.110282	100,000
FACSS [69]	NA	NA	NA	NA	NA	NA	NA	2996.37524	3006.41975	2999.413798	4.82E+00	NA
SNS [69]	3.5	0.7	17	7.3	7.71532	3.35021	5.28665	2994.47107	2994.4711	2994.47107	7.00E−06	3750



(a) Best Position **(b)** Convergence curve

Fig. 16 Convergence curve for SRD

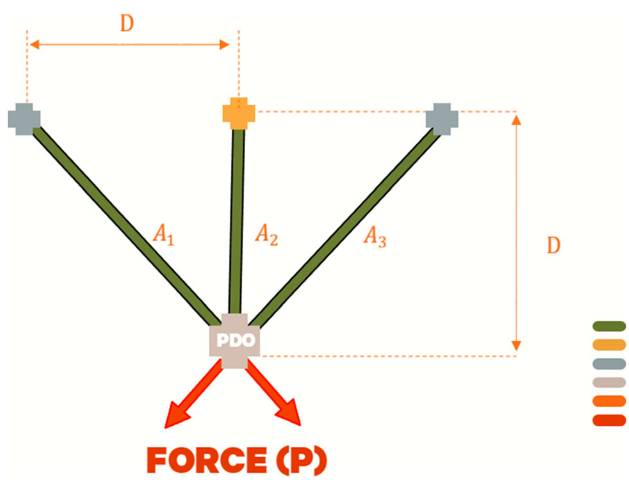


Fig. 17 Schematic illustration of the 3-BTD

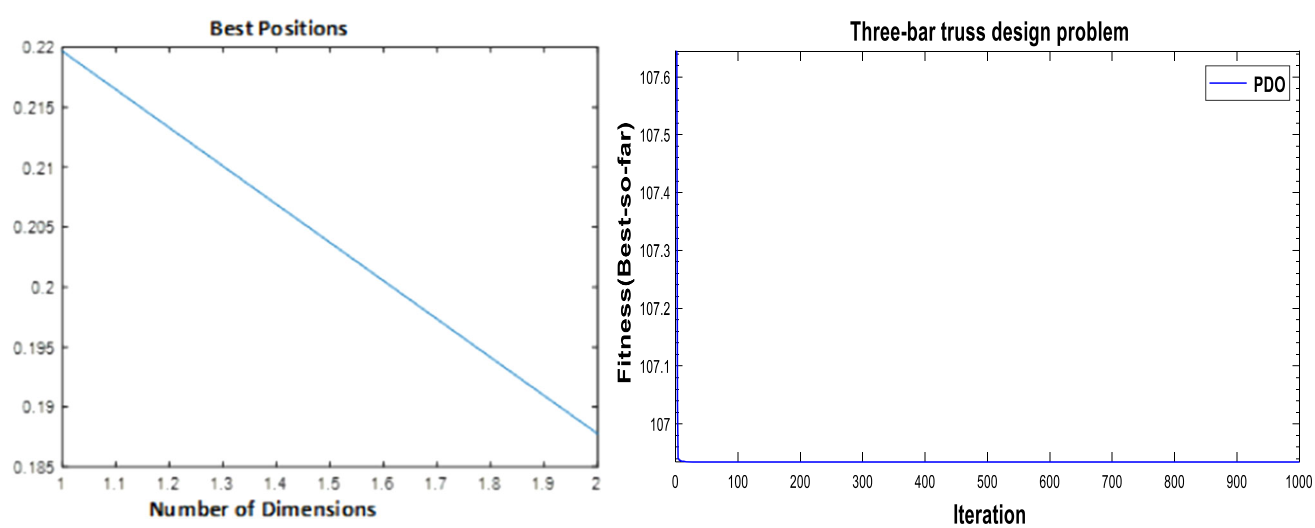
$$
\begin{aligned}
g_1(X) &= \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0, g_2(X) \\
&= \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0, g_3(X) \\
&= \frac{1}{\sqrt{2}x_2 + x_1} P - \sigma \leq 0
\end{aligned}
$$

$l = 100cm.P = 2kN/cm^3$, $\sigma = 2kN/cm^3$, range: $0 \leq x_1, x_2 \leq 1$ [75]

The PDO is applied to solve the 3-BTD problem. Its results are compared with some optimization algorithms in the literature that have been used to solve the same problem. The comparison is given in Table 12. It can be seen from the results that the PDO and nine other metaheuristic algorithms returned the best (least) solution; however, the mean value for the PDO is the least and so confirmed the superiority of the proposed PDO. Also, the qualitative results shown in Fig. 18 show that the PDO searched promising regions through the iterations because it steadily converged towards the optimal solution. The best position figure also showed that the PDO balanced the exploration and exploitation phases.

Table 12 Comparative results for 3-BTD

Algorithm	x1	x2	Best	Worst	Average	SD	FEs
PDO	0.219498412	0.188423865	**106.93**	106.93	106.93	**2.30E−14**	2,000
DMOA [58]	0.219515103	0.188372818	**106.93**	106.93	106.93	4.68E−05	5,000
AOA [58]	0.219498412	0.188372818	**106.93**	107.1	106.97	0.037732	20,000
CPSOGSA [58]	0.224256176	0.19245187	107.16	130.11	111.79	5.2175	5,000
PSO [58]	0.224256176	0.188423865	**106.93**	106.93	106.93	2.95E−14	20,000
BBO [27]	0.219498412	0.188372818	**106.93**	106.93	106.93	1.23E−13	50,000
DE [27]	0.224256176	0.19245187	**106.93**	106.93	106.93	3.23E−14	10,000
SSA [27]	0.78867	0.407569	**106.93**	106.93	106.93	4.83E−14	6,000
SCA [58]	0.224256176	0.19245187	**106.93**	106.94	106.94	0.00085771	10,000
GWO [58]	0.219498412	0.188423865	**106.93**	106.93	106.93	2.65E−07	5,000
GA [67]	0.788915	0.407569	263.895886	264.820805	263.968037	1.66862E−01	50,000
ICA [67]	0.788625	0.408389	263.895845	263.914133	263.899327	4.11693E−03	50,000
CS [67]	0.78867	0.40902	263.97156	NA	264.0669	9.00000E−05	15,000
WCA [72]	0.788651	0.408316	263.895843	263.896201	263.895903	8.71000E−05	5250
WSA [40]	0.788683	0.408227	263.895843	263.897432	263.896067	3.11960E−04	50,000
CGO [39]	NA	NA	263.895843	263.896007	263.895851	2.51E−05	100,000
AOS [67]	NA	NA	263.895843	263.895845	263.895843	8.26E−09	100,000
SNS [69]	0.78868473	0.4082211	263.895843	263.895856	263.895846	3.31056E−06	4800



(a) Best position **(b)** Convergence curve

Fig. 18 Convergence curve for the 3-BTD

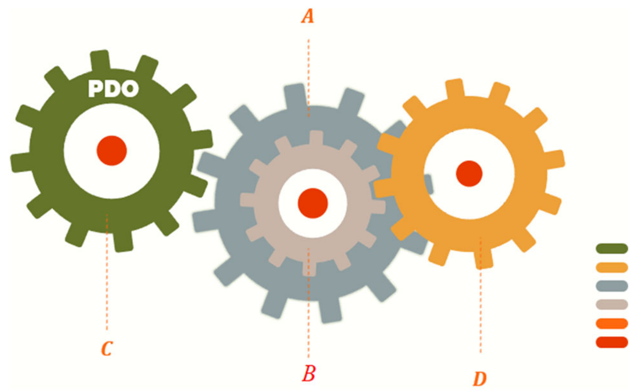


Fig. 19 The schematic diagram of the GTD

4.2.6 The gear train design problem (GTD)

The gear train design problem (GTD) is an unconstrained discrete design optimization problem first introduced by [70]. The GTD falls under the mechanical engineering domain. Minimizing the ratio of the output/input shaft's angular velocity is the design goal of the GTD, as shown in Fig. 19. The design variables are the number of teeth of gears given as $n_A(= x_1), n_B(= x_2), n_C(= x_3), n_D(= x_4)$. The GTD is mathematically modelled in Eq. 20.

$$\min f(X) = \left(\frac{1}{6.931} - \frac{x_3 x_2}{x_1 x_4}\right)^2 \tag{20}$$

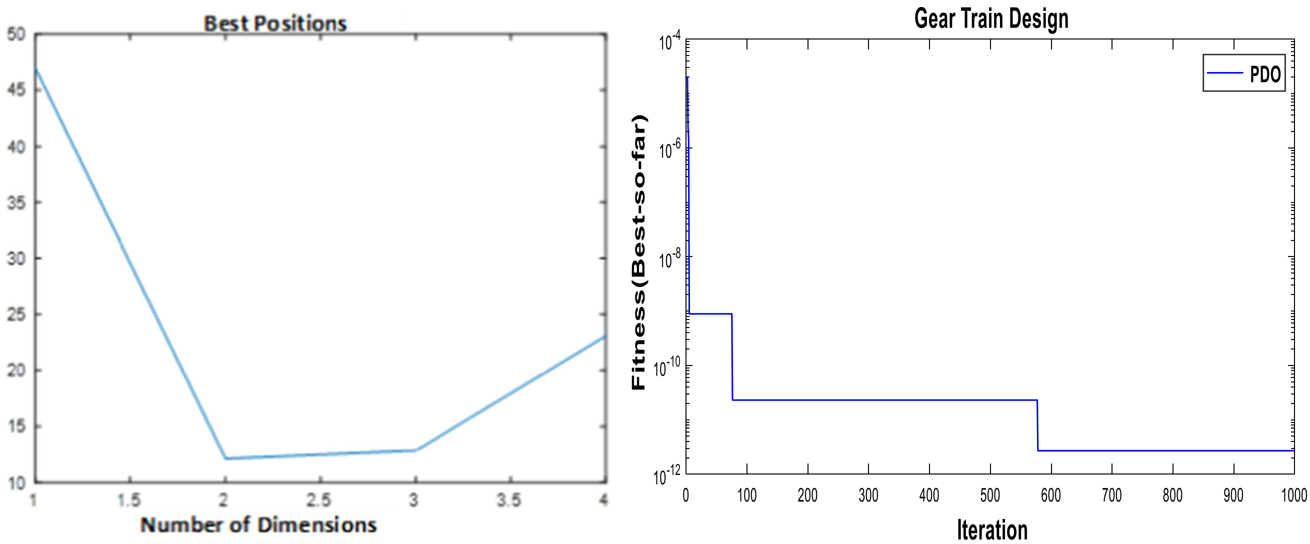
where

$x_1, x_2, x_3, x_4 \in \{12, 13, 14, \ldots, 60\}$ [70].

The best results of PDO and several other competitive algorithms that have been used to solve this problem are presented in Table 13. Though all the algorithms found the optimal solution as the 'best' value, the PDO proved to be a competitive, stable, and robust optimizer. Also, the qualitative results show that the PDO searched promising regions through the iterations because it steadily converged towards the optimal solution. The best position figure also showed that the PDO balanced the exploration and exploitation phases, as shown in Fig. 20.

Table 13 Comparative result for GTD

Algorithm	x1	x2	x3	x4	Best	Worst	Average	SD	FEs
PDO	48	17	22	54	2.70E−12	2.27E−08	1.14E−12	1.07E−11	5,000
DMOA [58]	49	19	16	43	2.70E−12	2.31E−11	6.78E−12	8.59E−12	20,000
AOA [58]	49	19	19	54	2.70E−12	2.73E−08	8.25E−09	1.23E−08	50,000
CPSOGSA [58]	55	16	16	43	2.31E−11	2.73E−08	3.45E−09	9.00E−09	50,000
PSO [58]	34	13	20	53	2.70E−12	2.36E−09	1.25E−09	1.04E−09	50,000
BBO [27]	53	26	15	51	2.70E−12	7.76E−08	5.26E−09	1.62E−08	50,000
DE [27]	43	16	34	43	2.70E−12	3.82E−09	1.02E−09	1.08E−09	50,000
SSA [27]	49	19	19	49	2.70E−12	1.12E−08	3.72E−10	2.04E−09	50,000
SCA [58]	49	19	34	49	2.70085714E−12	2.36E−09	1.24E−09	7.05E−10	25,000
GWO [58]	49	19	16	43	2.70E−12	2.31E−11	1.70E−11	9.84E−12	
GA [67]	49	19	16	43	2.70E−12	1.5247E−08	1.6212E−09	3.2174E−09	50,000
ICA [67]	43	16	19	49	2.70E−12	2.3576E−09	8.0417E−10	7.7862E−10	50,000
CS [67]	43	16	19	49	2.70E−12	6.51E−09	9.6633E−10	6.4529E−10	50,000
ABC [72]	49	16	19	43	2.70E−12	1.3616E−09	1.6800E−10	4.5748E−09	50,000
MSFWA [76]	49	19	16	43	2.70E−12	1.36165E−09	1.68012E−10	7.2953E−08	50,000
MBA [76]	43	16	19	49	2.70E−12	NA	NA	NA	NA
NNA [76]	49	16	19	43	2.70E−12	NA	NA	NA	NA
ISA [72]	43	19	16	49	2.70E−12	NA	NA	NA	NA
APSO [76]	43	16	19	49	2.70E−12	NA	NA	NA	NA
IAPSO [69]	43	16	19	49	2.70E−12	NA	NA	NA	NA
MVO [76]	43	16	19	49	2.70E−12	NA	NA	NA	NA
MFO [76]	43	19	16	49	2.70E−12	NA	NA	NA	NA
ALO [69]	49	19	16	43	2.70E−12	NA	NA	NA	NA
PSOSCALF [73]	49	19	16	43	2.70E−12	NA	NA	NA	NA
WSA [39]	43	16	19	49	2.70E−12	NA	NA	NA	NA
SNS [69]	43	19	16	49	2.70085714E−12	1.36165E−09	1.68012E−10	3.74894E−10	25,000



(a) Best position **(b)** Convergence curve

Fig. 20 Qualitative result for the GTD

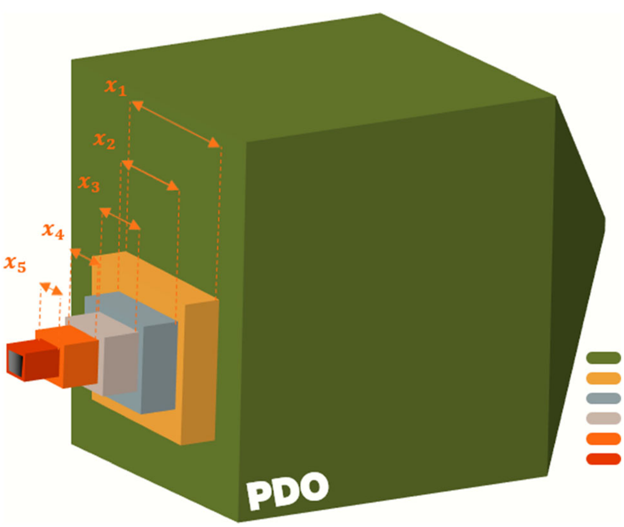


Fig. 21 Schematic diagram of the CBD

4.2.7 The cantilever beam design problem (CBD)

The proposed PDO is applied to solve the cantilever beam design problem. The cantilever is a square cross section, as shown in Fig. 21. The optimization aims to minimize a cantilever beam's weight given the following decision variables: the height or width of the five hollow square blocks whose thickness is constant [77]. The CBD is modelled mathematically as in Eq. 21.

$$\min f(X) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5) \tag{21}$$

Subject to

$g(X) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 00.01 \ \leq \ x_i \ \leq 100\forall$ $1 = 1, \ldots, 5$ [77].

Table 14 shows the values returned by each of the considered metaheuristic algorithms of the five design variables of the CBD, along with the optimal weight value.

Table 14 Comparative results for the CBD

Algorithm	x1	x2	x3	x4	x5	Best	Worst	Average	SD	FEs
PDO	5.689588272	5.020754741	4.261692336	3.312993832	2.040862644	**1.3004**	1.4048	1.3105	5.82E−05	4,000
DMOA [58]	5.694296874	5.025254683	4.253985946	3.314226276	2.037547058	1.3006	1.302	1.3009	0.00037768	15,000
AOA [58]	6.01513	5.025254683	4.253985946	3.312993832	2.037547058	1.3685	4.54	1.9597	0.67607	15,000
CPSOGSA [58]	5.689588272	4.418889751	4.119344796	3.314226276	2.15278	**1.3004**	1.3004	**1.3004**	3.50E−16	100,000
PSO [58]	5.694296874	5.30344	4.49587	7.754705773	2.15428	**1.3004**	1.3004	**1.3004**	2.37E−16	100,000
BBO [27]	6.01878	5.025254683	4.261692336	3.312993832	2.15428	**1.3004**	1.3004	**1.3004**	4.73E−14	100,000
DE [27]	6.01513	4.418889751	4.253985946	3.314226276	2.040862644	**1.3004**	1.3004	**1.3004**	2.26E−16	12,000
SSA [27]	5.689588272	5.020754741	4.119344796	7.754705773	2.037547058	**1.3004**	1.3004	**1.3004**	1.72E−12	50,000
SCA [58]	5.694296874	5.025254683	4.49587	3.49896	2.15278	1.3076	1.3394	1.3212	0.0085035	10,000
GWO [58]	6.322849342	4.418889751	4.495	3.50142	2.15428	1.3004	1.3004	1.3004	1.30E−07	50,000
SOS [76]	6.01878	5.30344	4.49587	3.49896	2.15564	1.33996	NA	1.33997	1.1E−5	15,000
CGO [69]	6.01513	5.3093	4.495	3.50142	2.15278	1.33997	1.340602	1.340052	1.2245E−04	100,000
AOS [70]	NA	NA	NA	NA	NA	1.339957	1.491711	1.351954	0.02499743	100,000
MGA [76]	NA	NA	NA	NA	NA	1.339976	1.340201	1.340053	6.99E−05	100,000
SNS [69]	6.01545	5.31066	4.488	3.50528	2.15428	1.339958	1.339958	1.339958	1.1102E−15	12,000

It can be observed that most of the algorithms returned the optimal solution; however, the PDO showed competitiveness in terms of the average weight value compared to the other algorithms. Figure 22 shows the qualitative (best position and convergence curve) of PDO, from which it can be noticed that the convergence curve has stepwise behaviour. The best position trajectory starts with exploration and then the exploitation phase to find the solution in the feasible region.

4.2.8 The optimal design of I-shaped beam (IBD)

The I-beam design problem (IBD) is an optimization problem that minimizes the vertical deflection of a beam. The IBD is formulated in Eq. 22, and the schematic diagram is shown in Fig. 23. The IBD is subject to two design constraints: the load's cross-sectional area and stress. The IBD four design variables are as follows: the flange's width $b(=x_1)$, the height of section $h(=x_2)$, the web's thickness $t_w(=x_3)$, and the flange's thickness $t_f(=x_4)$ [78].

$$\min f(X) = \frac{500}{\frac{x_3(x_2-2x_4)^3}{12} + \left(\frac{x_1 x_4^3}{6}\right) + 2bx_4(x_2-x_4)^2} \tag{22}$$

Subject to

$$g_1(X) = 2x_1x_3 + x_3(x_2-2x_4) \le 300, g_2(X) = \frac{18x_2 \times 10^4}{x_3(x_2-2x_4)^3 + 2x_1x_3\left(4x_4^2 + 3x_2(x_2-2x_4)\right)} + \frac{15x_1 \times 10^3}{(x_2-2x_4)x_3^2 + 2x_3x_1^3} \le 56$$

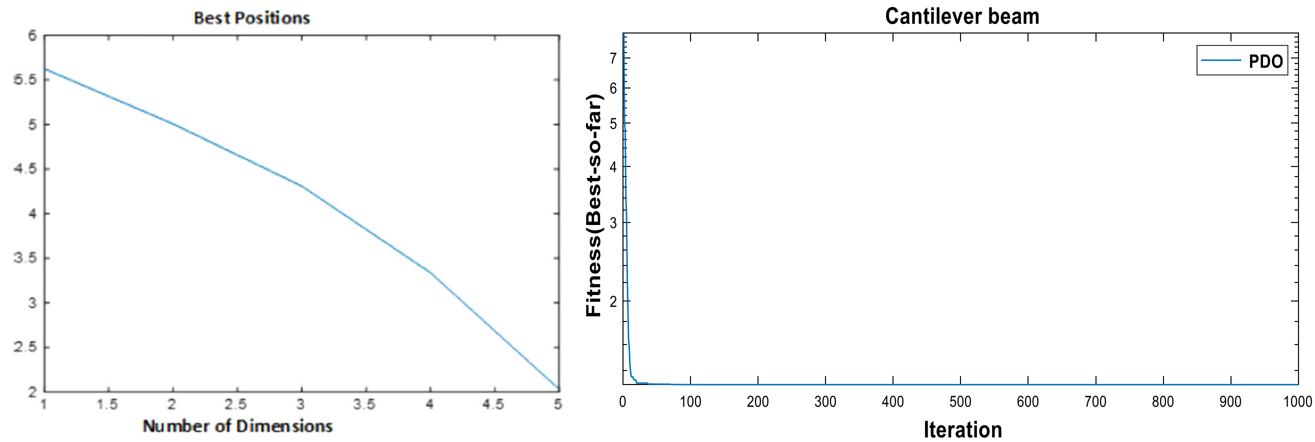
Range

$10 \le x_1 \le 50, 10 \le x_2 \le 80, 0.9 \le x_3 \le 5, 0.9 \le x_4 \le 5$ [78].

The values of the four design variables of the CBD, along with the optimal weight value, returned by each of the considered metaheuristic algorithms are shown in Table 15. It can be observed that most of the algorithms returned the optimal solution; however, the PDO is second to both SSA and GWO, which returned the least average weight value compared to the other competing algorithms. Figure 24 shows the qualitative (best position and convergence curve) of PDO, from which it can be noticed the convergence curve has stepwise behaviour. The best position trajectory starts with exploration and then the exploitation phase to find the solution in the feasible region.

4.2.9 The tubular column design (TCD)

The tabular column problem introduced [79] is an optimization problem that aims to find the minimum cost of a



(a) Best position **(b)** Convergence curve

Fig. 22 Convergence curve for CBD

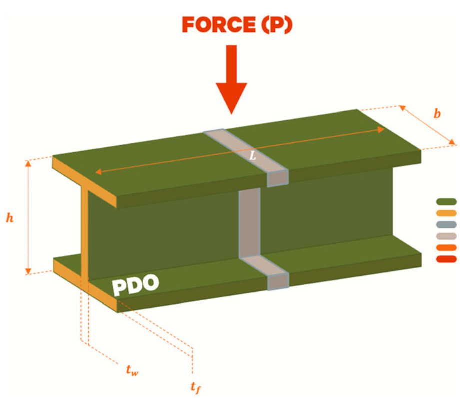


Fig. 23 The schematic diagram of the IBD

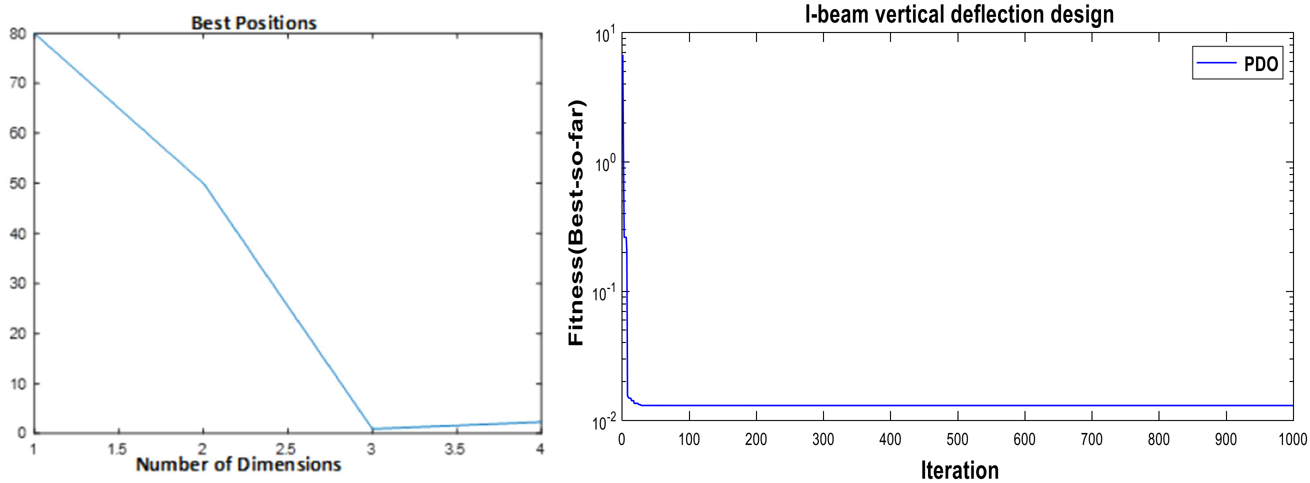
uniform column of the tubular section to carry a compressive load. There are two design variables are, namely: the TCD's mean diameter $d(=x_1)$ and the tube's thickness $t(=x_2)$. The schematic diagram of the TCD is shown in Fig. 25. The relevant literature postulates that the material used for the column has yielded stress $(\sigma_y = 500kgf/cm^2)$ and modulus of elasticity $\left(E = 0.85 \times 10^6 kgf/cm^2\right)$. The mathematically model of TCD is given in Eq. 23.

$$\min f(X) = 9.8x_1x_2 + 2x_1 \tag{23}$$

Subject to

Table 15 Comparative results for the IBD

Algorithm	x1	x2	x3	x4	Best	Worst	Average	SD	FEs
PDO	80	50	0.9	2.321795336	0.013074	0.013082	0.013075	1.54E−06	1,500
DMOA [58]	80	50	0.9	2.321792508	0.013074	0.013081	0.013075	1.95E−06	2,500
AOA [58]	80	50	0.9	2.321795336	0.013074	0.013162	0.013091	1.89E−05	1,950
CPSOGSA [58]	80	50	0.9	2.321792508	0.013074	0.013575	0.013203	0.00013379	2,500
PSO [58]	80	50	0.9	2.3217	− 7.64E+19	− 3.81E+09	− 1.40E+19	2.13E+19	4,000
BBO [27]	80	50	0.9	2.32	− 4.53E+13	− 6.56E+09	− 6.24E+12	1.24E+13	2,500
DE [27]	80	50	0.9	2.3216	− 1.71E+10	− 1.25E+08	− 2.35E+09	3.35E+09	4,000
SSA [27]	80	50	0.9	2.321795336	0.013074	0.013076	**0.013074**	5.44E−07	1,550
SCA [58]	80	50	0.9	2.321792508	0.013075	0.013191	0.013102	2.46E−05	2,500
GWO [58]	80	50	0.9	2.3216	0.013074	0.013075	**0.013074**	1.98E−07	4,000
EMGO-FCR [76]	80	50	0.9	2.32	0.0131	NA	NA	NA	NA
CS [76]	80	50	0.9	2.3216	0.013075	0.0135365	0.013217	0.0001345	5000
SOS [69]	80	50	0.9	2.3217	0.013074	NA	0.013088	4.0E−5	5000
AOS [69]	NA	NA	NA	NA	0.013074	0.013814	0.013179	1.555E−04	100,000
SNS [69]	80	50	0.9	2.3217	0.013074	0.0130764	0.013074	4.313E−07	3600



(a) Best position **(b)** Convergence curve

Fig. 24 Convergence curve for IBD

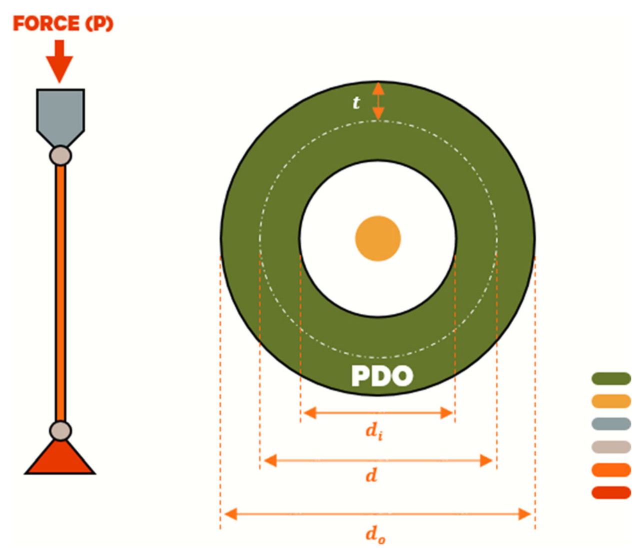


Fig. 25 Schematic illustration of TCD

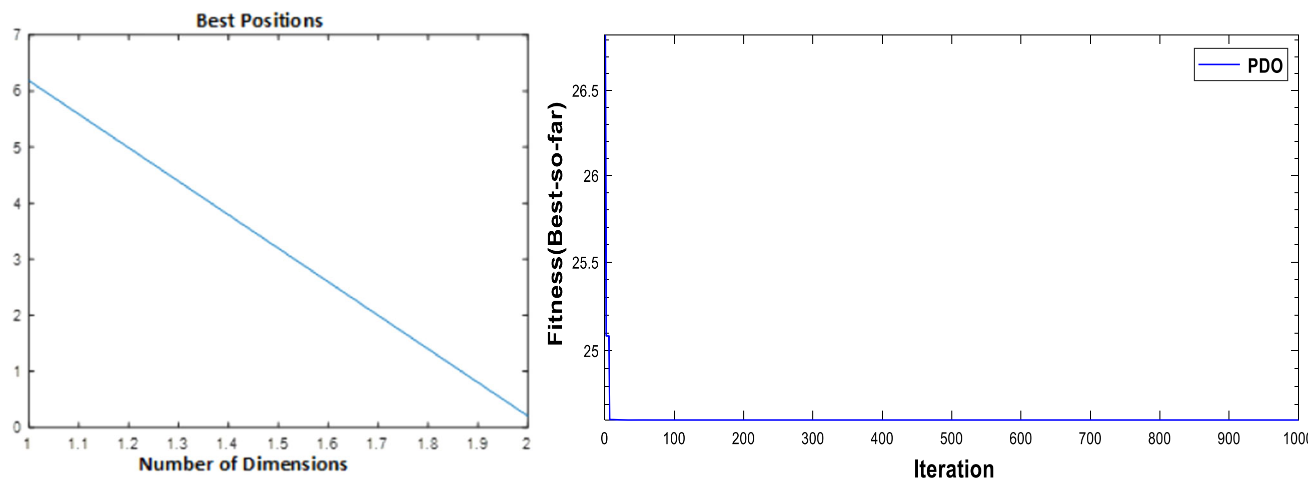
$$g_1(X) = \frac{P}{\pi x_1 x_2 \sigma_y} - 1 \leq 0, g_2(X) = \frac{8PL^2}{\pi^3 E x_1 x_2 \left(x_1^2 + x_2^2\right)} - 1 \leq 0, g_3(X) = \frac{2.0}{x_1} - 1 \leq 0, g_4(X) = \frac{x_1}{14} - 1 \leq 0,$$

$g_5 = \frac{0.2}{x_2} - 1 \leq 0, g_6(X) = \frac{x_2}{8} - 1 \leq 0, where 2 \leq x_1 \leq 14, 0.2 \leq x_2 \leq 0.8$ [79].

So many algorithms available in the literature have been used to solve the TCD problem, and some of the previous results are compared with the PDO's result and given in Table 16. Six algorithms, including PDO, returned the least cost function value. Four algorithms, including the PDO, had the smallest average value; however, the PDO returned its result in the least number of iterations corresponding to 500 function evaluations. Figure 26 shows the qualitative (best position and convergence curve) of PDO. It can be noticed that the convergence curve has stepwise behaviour,

Table 16 Comparative results for TCD

Algorithm	x1	x2	Best	Worst	Average	SD	FEs
PDO	6.182678144	0.2	**24.615**	24.615	**24.615**	8.84E−10	500
DMOA [58]	6.182683216	0.2	**24.615**	24.615	**24.615**	8.80E−06	4,000
AOA [58]	5.45115623	0.29196547	24.616	24.688	24.632	0.019741	3000
CPSOGSA [58]	5.45139	0.29196	24.615	24.615	**24.615**	1.81E−14	15,000
PSO [58]	5.45115623	0.2	2183	2183	2183	9.25E−13	3000
BBO [27]	6.182683216	0.2	2183	2183	2183	9.25E−13	100,000
DE [27]	6.179782123	0.2	2183	2183	2183	9.25E−13	5,000
SSA [27]	5.45115623	0.29196547	24.615	24.615	**24.615**	6.53E−12	10,000
SCA [58]	5.45139	0.29196	24.615	24.616	24.616	0.00027979	6,000
GWO [58]	6.182683216	0.29196	24.615	24.615	24.615	7.62E−07	10,000
ISA [72]	5.45115623	0.29196547	26.5	2.65E+01	26.531	NA	3000
CS [76]	5.45139	0.29196	26.53217	26.53972	26.53504	1.93E−03	15,000
FA [73]	NA	NA	26.52	NA	28.74	2.08	3000
AOS [69]	NA	NA	26.5313783	26.6083136	26.531614	1.0300E−03	100,000
SNS [69]	5.45115623	0.29196547	26.4863615	26.486371	26.4863625	2.2160E−06	1250



(a) Best position (b) Convergence curve

Fig. 26 Convergence curve for TCD

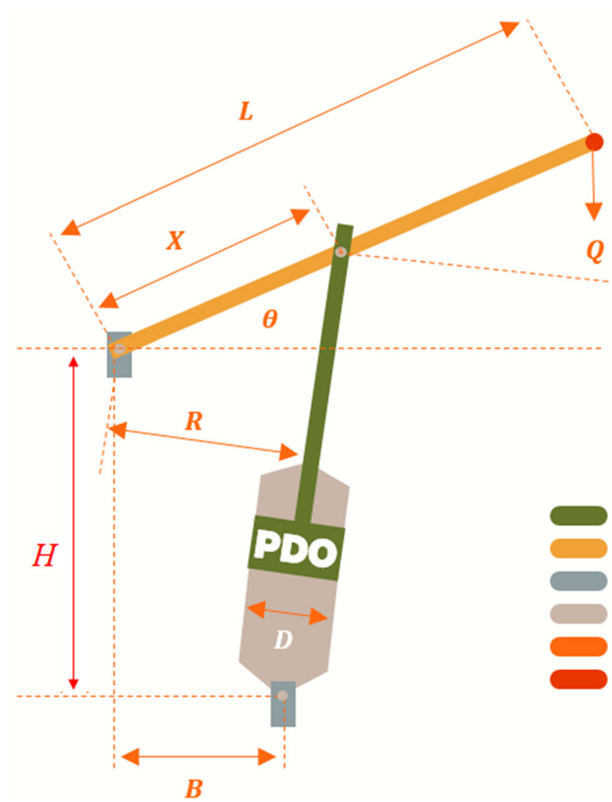


Fig. 27 Schematic illustration of the PLD

and the PDO converged after the second iteration. The best position trajectory starts with exploration and then the exploitation phase to find the solution in the feasible region.

4.2.10 The piston lever design problem (PLD)

The piston lever design (PLD) is an optimization problem that aims to minimize the volume of oil when the piston's lever is lifted from $0^o to 45^o$. There are four design components whose location is crucial; the components are as follows $H(=x_1), B(=x_2), D(=x_3), and X(=x_4)$. The PLD is as shown in Fig. 27, and the mathematical model is given in Eq. 24 [78].

$$\min f(X) = \frac{1}{4}\pi x_3^2 (L_2 - L_1) \tag{24}$$

Subject to

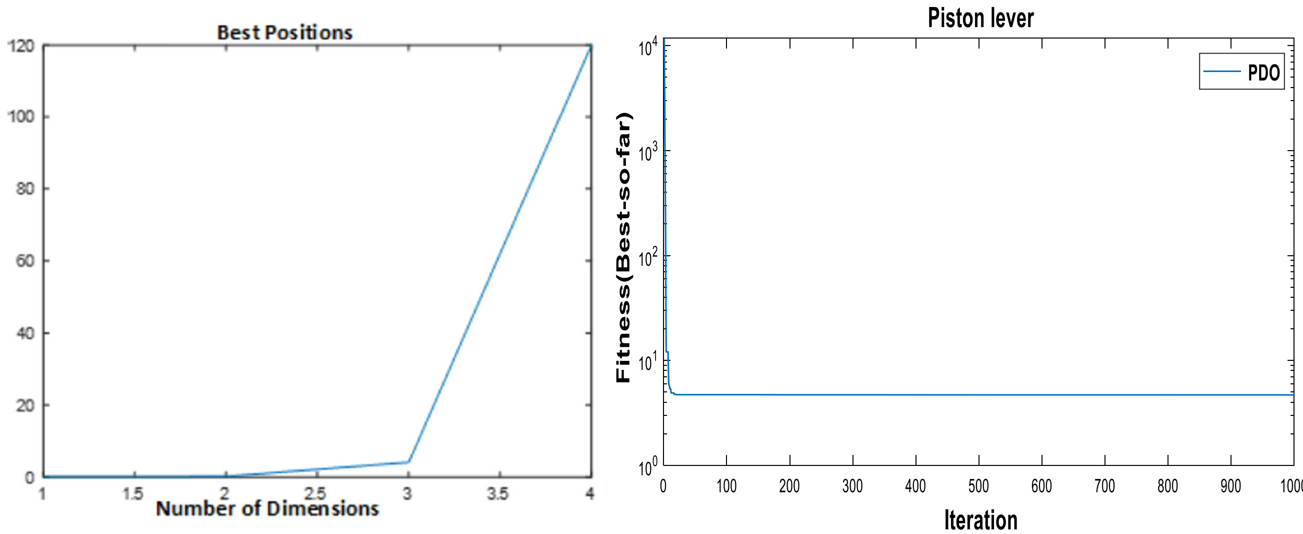
$$\begin{aligned} g_1(X) &= QLcos\theta - R \times F \leq 0, g_2(X) \\ &= Q(L - x_4) - M_{max} \leq 0, g_3(X) \\ &= 1.2(L_2 - L_1) - L_1 \leq 0, \end{aligned}$$

$$g_4(X) = \frac{x_3}{2} - x_2 \leq 0$$

where

Table 17 Comparative result for PLD

Algorithm	x1	x2	x3	x4	Best	Worst	Average	SD	FEs
PDO	0.05	0.144897318	4.11572157	120	**4.602**	4.7728	**4.7222**	0.015037	3,000
DMOA [58]	0.05	0.125073578	4.116042166	120	4.695	4.7054	4.698	0.0026177	5,000
AOA [58]	0.05	0.125073578	4.116042166	120	7.378	483.32	310.42	129.39	50,000
CPSOGSA [58]	500	500	2.578147082	120	4.6949	537.36	231.85	140.81	50,000
PSO [58]	133.3	2.44	117.14	4.75	4.6949	4.6949	**4.6949**	1.67E−08	50,000
BBO [27]	129.4	2.43	119.8	4.75	4.6956	4.7153	4.7002	0.0043076	50,000
DE [27]	250	3.96	60.03	5.91	4.6949	4.7006	4.7	0.0017429	50,000
SSA [27]	135.5	2.48	116.62	4.75	4.6949	5.8417	4.9188	0.32217	50,000
SCA [58]	0.05	0.144897318	4.11572157	120	4.6977	4.725	4.7071	0.0080795	12,500
GWO [58]	0.05	0.125073578	4.116042166	120	4.6949	167.63	42.704	70.074	100,000
GA [71]	250	3.96	60.03	5.91	161	216	185	18.2	50,000
HPSO [71]	135.5	2.48	116.62	4.75	162	197	187	13.4	50,000
HPSO with Q-learning [71]	NA	NA	NA	NA	129	168	151	13.4	50,000
CS [71]	0.05	2.043	120	4.085	8.4271	168.592	40.2319	59.0552	50,000
ISA [71]	NA	NA	NA	NA	8.4	610.6	226.5	111.2	12,500
CGO [72]	NA	NA	NA	NA	8.41281381	167.472809	45.04866	67.24763	100,000
AOS [76]	NA	NA	NA	NA	8.41914274	167.664986	33.741276	93.4667472	100,000
MGA [69]	NA	NA	NA	NA	8.41340665	167.473213	32.468893	29.9637044	100,000
SNS [69]	0.05	2.042	120	4.083	8.41269835	167.472775	24.318974	47.7179265	5000



(a) Best position **(b)** Convergence curve

Fig. 28 Convergence curve for PLD

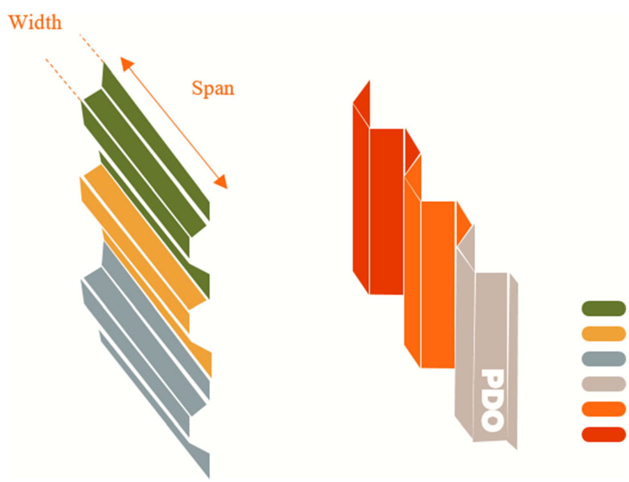


Fig. 29 Schematic illustration of the CBhD problem

$$R = \frac{|-x_4(x_4 sin\theta + x_1) + x_1(x_2 - x_4 cos\theta)|}{\sqrt{(x_4 - x_2)^2 + x_1^2}}, F = \frac{\pi P x_3^2}{4}, L_1 = \sqrt{(x_4 - x_2)^2 + x_1^2},$$

$$L_2 = \sqrt{(x_4 sin\theta + x_1)^2 + (x_2 - x_4 cos\theta)^2}, \theta = 45^0, Q = 10,000 lbs, L = 240 in, M_{max} = 1.8 \times 10^6 ibsin, P = 1500 psi,$$

Range: $0.05 \le x_1, x_2, x_4 \le 500, 0.05 \le x_3 \le 120$ [78].

Looking at Table 17, the PDO returned the least optimal cost of the objective function compared to other

Table 18 Comparative result for CBhD

Algorithm	x1	x2	x3	x4	Best	Worst	Average	SD	FEs
PDO	48.31191	54.78270401	61.92983	0.424913	6.9821	7.1475	6.9949	0.93079	500
DMOA [58]	48.13864556	5.45E+01	62.04388433	0.43144458	4.6949	4.7006	4.6996	0.002165	4,000
AOA [58]	57.69277	34.13296	57.55294	1.05007	481.97	481.97	481.97	0	3000
CPSOGSA [58]	48.31191	54.78270401	61.92983	0.424913	4.6949	571.24	232.81	149.31	15,000
PSO [58]	37.1179498	33.035021	37.1939476	0.7306255	4.6949	4.6949	4.6949	2.33E−08	3000
BBO [27]	57.69231	34.14762	57.69231	1.05	2.79E+12	2.79E+12	2.79E+12	0.000497	100,000
DE [27]	57.69277	34.13296	57.55294	1.05007	4.6949	6.5845	5.0255	0.48808	5,000
SSA [27]	48.31191	54.78270401	61.92983	0.424913	4.6971	4.7544	4.7112	0.012761	10,000
SCA [58]	48.13864556	5.45E+01	62.04388433	0.43144458	4.695	167.67	37.278	66.276	6,000
GWO [58]	43.46368343	74.91894765	100	0.386989528	4.6689	4.6702	4.6692	0.00028189	10,000
FA [72]	37.1179498	33.035021	37.1939476	0.7306255	7.21	NA	10.23	1.95	12,000
LF-FA [72]	57.69231	34.14762	57.69231	1.05	6.95	NA	8.83	1.26	12,000
LS-LF-FA [72]	57.69277	34.13296	57.55294	1.05007	6.86	NA	7.44	0.67	12,000
AD-IFA [69]	NA	NA	NA	NA	6.84	NA	7.21	0.58	12,000
AOS [72]	NA	NA	NA	NA	6.84295801	7.06693619	7.06080838	6.4911E−04	100,000
SNS [69]	57.69230732	34.14762029	57.69230729	1.05	6.84296052	6.8430744	6.8429798	2.0942E−05	3125

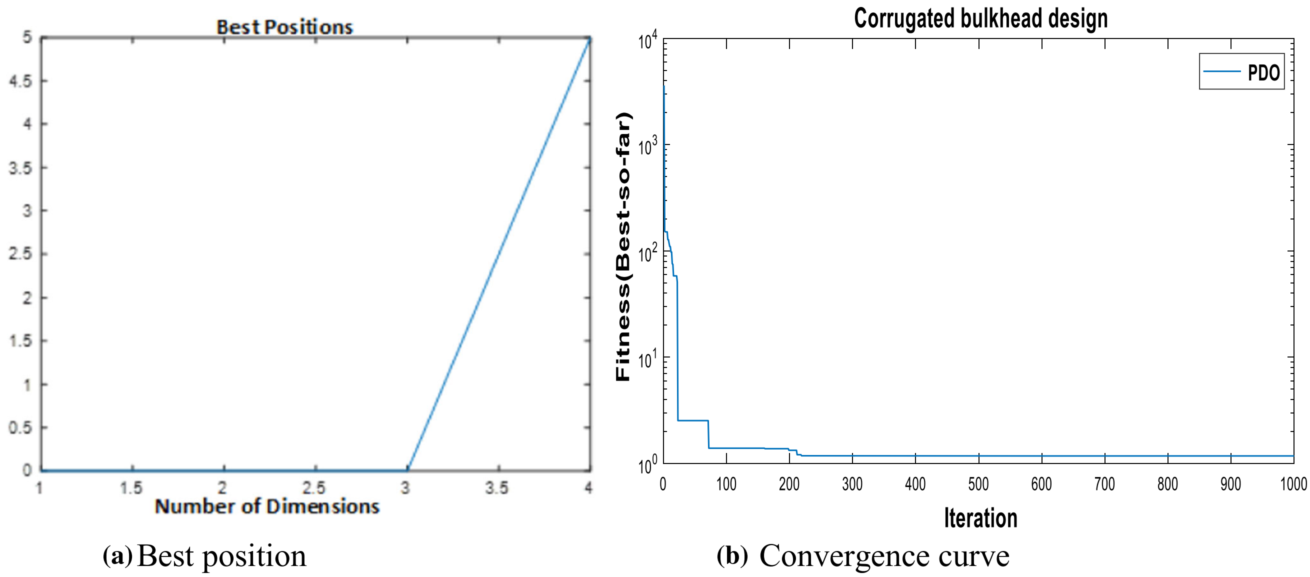


Fig. 30 Convergence curve of CBhD

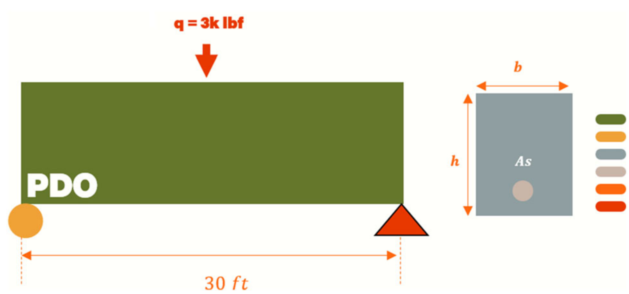


Fig. 31 Schematic illustration of the RCB

algorithms. The mean cost value and standard deviation showed that PDO has stability in the results returned.

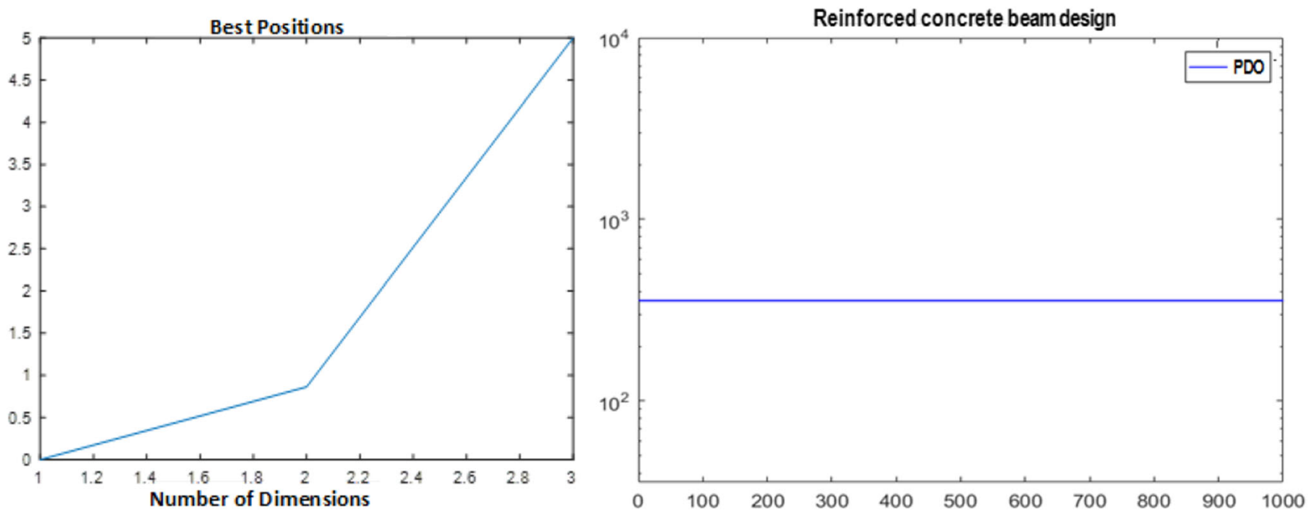
Figure 28 shows the qualitative (best position and convergence curve) of PDO. It can be noticed that the convergence curve has a steady stepwise convergence towards the optimal solution, and the PDO converges after the second iteration. The best position trajectory starts with exploration and then the exploitation phase to find the solution in the feasible region.

4.2.11 The corrugated bulkhead design problem (CBhD)

The goal of the corrugated bulkhead design is to minimize the weight of the chemical tanker's corrugated bulkhead. There are four design variables, namely: the width $(= x_1)$,

Table 19 Comparative result for RCB

Algorithm	x1	x2	x3	Best	Worst	Average	SD	FEs
PDO	6.32	33	8	357.4	357.4	**357.4**	1.17E−13	100
DMOA [58]	6	30	5	357.5	357.5	357.5	2.12E−10	150
AOA [58]	6	37	5	357.6	357.6	357.6	3.17E−09	25,000
CPSOGSA [58]	8.4	40	5	357.6	357.6	357.6	5.17E−12	5000
PSO [58]	7.2	30	5	357.6	357.6	357.6	7.12E−11	100,000
BBO [27]	7	20	5	357.6	357.6	357.6	9.12E−10	1000
DE [27]	6	20	10	357.6	357.6	357.6	8.12E−10	1000
SSA [27]	6	28	5	357.6	357.6	357.6	8.12E−08	500
SCA [58]	7.8	40	5	357.6	357.6	357.6	4.92E−12	25,000
GWO [58]	6.6	30	9	357.6	357.6	357.6	1.66E−08	1000
FA [72]	6.32	34	9	359.21	669.15	460.706	80.7387	25,000
CS [76]	6.32	34	9	359.21	NA	NA	NA	5000
AOS [72]	6.32	34	9	359.21	362.254	359.3306872	0.596149	100,000
SNS [69]	6.32	34	9	359.21	362.634	359.3222001	0.6149858	1000



(a) Best position **(b)** Convergence curve

Fig. 32 Convergence curve for RCB

depth $(= x_2)$, length $(= x_3)$, and thickness of the plate $(= x_4)$. The CBhD problem is illustrated in Fig. 29 and modelled in Eq. 25 [80].

$$\min f(X) = \frac{5.885x_4(x_1 + x_3)}{x_1 + \sqrt{|x_3^2 - x_2^2|}} \tag{25}$$

Subject to

$$g_1(X) = -x_4x_2\left(0.4x_1 + \frac{x_3}{6}\right) + 8.94\left(x_1 + \sqrt{|x_3^2 - x_2^2|}\right) \leq 0, g_2(X) = -x_4x_2^2\left(0.2x_1 + \frac{x_3}{12}\right) + 2.2\left(8.94\left(x_1 + \sqrt{|x_3^2 - c_2^2|}\right)\right)^{\frac{4}{3}},$$
$$g_3(X) = -x_4 + 0.0156x_1 + 0.15 \leq 0,$$
$$g_4(X) = -x_4 + 0.0156x_3 + 0.15 \leq 0,$$
$$g_5(X) = -x_4 + 1.05 \leq 0, g_6(X) = -x_3 + x_2 \leq 0$$

range: $0 \leq x_1, x_2, x_3 \leq 100, 0 \leq x_4 \leq 5$ [80].

The results of the application of the PDO and some other metaheuristic algorithms in solving the CBhD problem are shown in Table 18. It can be observed that the PDO was competitive with the other algorithms and was stable across the optimization process. As shown in Fig. 30, the qualitative result depicts how the PDO converges steadily during the optimization and balances the exploration and the exploitation as it solves the optimization problem.

4.2.12 The reinforced concrete beam design problem (RCB)

The reinforced concrete beam design problem, as shown in Fig. 31, consists of a beam supported by a span of about 30ft and subjected to both live and dead loads of about 2000lbf and 1000lbf, respectively [81]. It is assumed that the compressive strength of the concrete $(\sigma_c = 5ksi)$ reinforces steel yield stress $(\sigma_y = 50ksi)$. Three design variables are considered, namely: the reinforcement area $A_s(= x_1)$, the beam width $b(= x_2)$, and the depth of the beam $h(= x_3)$. The RCB is mathematically modelled in Eq. 26.

$$\min f(X) = 2.9x_1 + 0.6x_2x_3 \tag{26}$$

Subject to

$$g_1 = \frac{x_2}{x_3} - 4 \leq 0, g_2 = 180 + 7.375\frac{x_1^2}{x_3} - x_1x_2 \leq 0$$

Range: $x_1 \in \{6, 6.16, 6.32, 6.6, 7, 7.11, 7.2, 7.8, 7.9, 8, 8.4\}$, $x_2 \in \{28, 29, 30, \ldots, 40\}, 5 \leq x_3 \leq 10$ [81].

Looking at Table 19, we see that PDO returned the least objective function value for the RCB. The result returned by PDO significantly improves the results returned by other algorithms available in the literature. This significant improvement was achieved within the least possible iterations and function evaluations (100). The qualitative results are shown in Fig. 32, which shows how the PDO converged to the optimal solution and remained around the best value throughout the optimization process.

4.3 The pros and cons of the proposed PDO

We can see from the results presented in the previous subsection that the PDO demonstrated high capacity and capability to handle the different optimization problems considered: the classical CEC 2020 benchmark functions and the twelve optimization problems in the engineering domain. The results and statistical analysis confirm that the PDO is a robust and effective tool for optimization. The main advantages of the proposed PDO are as follows:

- Ease of implementation
- The PDO has a balanced exploration and exploitation, as seen from its performance in benchmark functions and real-world engineering problems.
- Fast and smooth convergence
- Few control parameters.

Although the proposed PDO's performance seems to be very promising, it still would require some further enhancement to be able to handle other real-world application problems different from all the engineering problems considered in this paper, which is in line with the "no free lunch" theory. Also, the fitness value returned at each iteration plays a dominant role in the updating mechanism. As such, a new updating mechanism may be considered for the PDO to improve its performance. Also, the PDO may be tested on more real-world problems like parallel machine scheduling, feature selection, and many more.

5 Conclusion and future works

This study proposes a novel natural-inspired population-based metaheuristic algorithm called the prairie dog optimization (PDO). The proposed algorithm mimics the foraging and burrowing activities and the specific response to the prairie dogs' unique alarms (communication). The foraging and burrowing activities are used for exploration, while the specific response to two unique alarms (antipredation and food source) is used for exploitation. The four activities were modelled mathematically to update the positions of the candidate solutions.

The performance of the proposed PDO in terms of exploration and exploitation was tested using the classical and CEC 2020 benchmark test functions. The results demonstrated the effectiveness of PDO in finding the optimal global solutions and having more steady convergence compared to other well-known optimization algorithms published in the literature. The statistical analysis was performed using the Freidman ranking test to establish the efficacy of the proposed PDO. The statistical results also confirmed the robustness of the proposed PDO in carrying out effective exploration and exploitation.

Furthermore, the proposed PDO was used to solve twelve real-world engineering problems and the results returned by the proposed PDO confirmed the capability of the algorithm to return better (near-optimal) solutions compared to some metaheuristic algorithms available in the literature. And PDO showed a high capacity to handle various constraints in the optimization problems.

The proposed PDO solved only single-objective continuous optimization problems, and researchers may consider developing the binary version of the algorithm. Also, the multi-objective variant of the PDO may be developed. Modifying and hybridizing the PDO may also be another direction researchers may exploit. Extending the PDO to handle other discrete or continuous real-world problems is a promising venture that researchers can undertake.

Declarations

Conflict of interest The authors declare no conflict of interest relating to this work.

References

1. Ezugwu AE (2021) Advanced discrete firefly algorithm with adaptive mutation-based neighborhood search for scheduling unrelated parallel machines with sequence-dependent setup times. Int J Intell Syst
2. Horst R, Tuy H (2013) Global optimization: deterministic approaches. Springer, New York
3. Abualigah L (2020) Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. Neural Comput Appl 25:1–24
4. Ezugwu AE, Shukla AK, Nath R, Akinyelu AA, Agushaka JO, Chiroma H, Muhuri PK (2021) Metaheuristics: a comprehensive overview and classification along with bibliometric analysis. Artif Intell Rev 87:1–80
5. Agushaka JO, Ezugwu AE (2021) Evaluation of several initialization methods on arithmetic optimization algorithm performance. J Intell Syst 31(1):70–94
6. Agushaka J, Ezugwu A (2020) Influence of initializing krill herd algorithm with low-discrepancy sequences. IEEE Access 8:210886–210909
7. Gardiner CW (1985) Handbook of stochastic methods, vol 3. Springer, Berlin
8. Agushaka JO, Ezugwu AE (2022) Influence of probability distribution initialization methods on the Performance of Advanced Arithmetic Optimization Algorithm with Application to Unrelated Parallel Machine Scheduling Problem. Concurr Comput Pract Exp
9. Dokeroglu T, Sevinc E, Kucukyilmaz T, Cosar A (2019) A survey on new generation metaheuristic algorithms. Comput Ind Eng 137:106040
10. Holland JH (1975) Adaptation in natural and artificial systems. University of Michigan Press, Michigan (second edition: MIT Press, 1992)
11. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol 4
12. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680
13. Dorigo M, Di Caro G (1999) Ant colony optimization: a new meta-heuristic. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), vol 2

14. Akay B, Karaboga D (2012) Artificial bee colony algorithm for large-scale problems and engineering design optimization. J Intell Manuf 23(4):1001–1014
15. Agushaka JO, Ezugwu AE (2022) Initialisation approaches for population-based metaheuristic algorithms: a comprehensive review. Appl Sci 12(2):896
16. Abualigah L, Diabat A (2021) Advances in sine cosine algorithm: a comprehensive survey. Artif Intell Rev 54:1–42
17. Ezugwu AE, Adeleke OJ, Akinyelu AA, Viriri S (2020) A conceptual comparison of several metaheuristic algorithms on continuous optimization problems. Neural Comput Appl 32(10):6207–6251
18. Ezugwu AE, Akutsah F (2018) An improved firefly algorithm for the unrelated parallel machines scheduling problem with sequence-dependent setup times. IEEE Access 6:54459–54478
19. Noshadi A, Shi J, Lee WS, Shi P, Kalam A (2016) Optimal PID-type fuzzy logic controller for a multi-input multi-output active magnetic bearing system. Neural Comput Appl 27(7):2031–2046
20. Abonyi J, Feil B (2007) Cluster analysis for data mining and system identification. Springer, Birkhäuser
21. Nguyen P, Kim JM (2016) Adaptive ECG denoising using genetic algorithm-based thresholding and ensemble empirical mode decomposition. Inf Sci 373:499–511
22. Oyelade ON, Ezugwu AE (2021) Characterization of abnormalities in breast cancer images using nature-inspired metaheuristic optimized convolutional neural networks model. Concurr Comput Pract Exp 84:e6629
23. Oyelade ON, Ezugwu AE (2021) A bioinspired neural architecture search based convolutional neural network for breast cancer detection using histopathology images. Sci Rep 11(1):1–28
24. Idris H, Ezugwu AE, Junaidu SB, Adewumi AO (2017) An improved ant colony optimization algorithm with fault tolerance for job scheduling in grid computing systems. PLoS ONE 12(5):e0177567
25. Ezugwu AE, Adeleke OJ, Viriri S (2018) Symbiotic organisms search algorithm for the unrelated parallel machines scheduling with sequence-dependent setup times. PLoS ONE 13(7):e0200030
26. Ezugwu AE (2019) Enhanced symbiotic organisms search algorithm for unrelated parallel machines manufacturing scheduling with setup times. Knowl-Based Syst 172:15–32
27. Agushaka JO, Ezugwu AE (2021) Advanced Arithmetic Optimization Algorithm for solving mechanical engineering design problems. PLoS ONE 16(8):e0255703
28. Abualigah L, AbdElaziz M, Sumari P, Geem ZW, Gandomi AH (2021) Reptile Search Algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert Syst Appl 191:116158
29. Kosorukoff A (2001) Human based genetic algorithm. In: 2001 IEEE international conference on systems, man and cybernetics. e-systems and e-man for cybernetics in cyberspace (Cat. No. 01CH37236)
30. Biswas A, Mishra K, Tiwari S, Misra A (2013) Physics-inspired optimization algorithms: a survey. J Optim 984:2013
31. Parpinelli RS, Lopes HS (2011) New inspirations in swarm intelligence: a survey. Int J Bio-Inspired Comput 3(1):1–16
32. Fogel DB (1998) Artificial intelligence through simulated evolution. Wiley, New York
33. Storn R, Price K (1997) Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359
34. Hansen N, Müller SD, Koumoutsakos P (2003) Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evol Comput 11(1):1–18
35. Połap D, Woźniak M (2021) Red fox optimization algorithm. Expert Syst Appl 166:114107
36. Abualigah L, Shehab M, Alshinwan M, Alabool H (2019) Salp swarm algorithm: a comprehensive survey. Neural Comput Appl 32:11195–11215
37. Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248
38. Hashim FA, Hussain K, Houssein EH, Mabrouk MS, Al-Atabany W (2021) Archimedes optimization algorithm: a new meta-heuristic algorithm for solving optimization problems. Appl Intell 51(3):1531–1551
39. Kaveh A, Dadras A (2017) A novel meta-heuristic optimization algorithm: thermal exchange optimization. Adv Eng Softw 110:69–84
40. Rao RV, Savsani VJ, Vakharia DP (2011) Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput Aided Des 43(3):303–315
41. Atashpaz-Gargari E, Lucas C (2007) Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: 2007 IEEE congress on evolutionary computation
42. Ghosh A, Das S, Mullick SS, Mallipeddi R, Das AK (2017) A switched parameter differential evolution with optional blending crossover for scalable numerical optimization. Appl Soft Comput 57:329–352
43. Ghambari S, Rahati A (2018) An improved artificial bee colony algorithm and its application to reliability optimization problems. Appl Soft Comput 62:736–767
44. Zhong F, Li H, Zhong S (2016) A modified ABC algorithm based on improved-global-best-guided approach and adaptive-limit strategy for global optimization. Appl Soft Comput 46:469–486
45. Sun G, Liu Y, Yang M, Wang A, Liang S, Zhang Y (2017) Coverage optimization of VLC in smart homes based on improved cuckoo search algorithm. Comput Netw 116:63–78
46. Peraza C, Valdez F, Garcia M, Melin P, Castillo O (2016) A new fuzzy harmony search algorithm using fuzzy logic for dynamic parameter adaptation. Algorithms 9(4):69
47. Wolpert DH, Macready WG (1997) No free lunch theorems for optimizations. IEEE Trans Evol Comput 1(1):67–82
48. Hygnstrom SE, Virchow DR (2002) Prairie dogs and the prairie ecosystem. Pap Nat Resour 36:3149
49. Long K (2002) Prairie dogs: a wildlife handbook. Johnson Books, Boulder
50. Hoogland JL (1995) The black-tailed prairie dog: social life of a burrowing mammal. University of Chicago Press, Chicago
51. Chance G (1976) Wonders of prairie dogs. Dodd, Mead, and Company, New York
52. Fitzgerald JP, Lechleitner RR (1974) Observations on the biology of Gunnison's prairie dog in central Colorado. Am Midl Nat 87:146–163
53. Mulhern DW, Knowles CJ (1997) Black-tailed prairie dog status and future conservation planning. In: Uresk DW, Schenbeck GL, O'Rourke JT (eds) Conserving Biodiversity on Native Rangelands: symposium proceedings: August 17, 1995, Fort Robinson State Park, Nebraska. General Technical Report RM-GTR-298. US Department of Agriculture, Forest Service, Rocky Mountain Forest and Range Experiment Station, Fort Collins, vol 298, pp 19–29
54. Slobodchikoff CN, Kiriazis J, Fischer C, Creef E (1991) Semantic information distinguishing individual predators in the alarm calls of Gunnison's prairie dogs. Anim Behav 42(5):713–719
55. Slobodchikoff CN, Perla BS, Verdolin JL (2009) Prairie dogs: communication and community in an animal society. Harvard University Press, Harvard
56. Slobodchikoff CN (2002) Cognition and communication in prairie dogs. In: Beckoff M, Allen C, Burghardt GM (eds) The cognitive animal: empirical and theoretical perspectives on animal cognition. A Bradford Book, Cambridge, pp 257–264

57. Yang XS, Deb S (2009) Cuckoo search via Lévy flights. In: 2009 World congress on nature & biologically inspired computing (NaBIC)
58. Agushaka JO, Ezugwu AE, Abualigah L (2022) Dwarf mongoose optimization algorithm. Comput Methods Appl Mech Eng 391:114570
59. Abualigah L, Diabat A, Mirjalili S, AbdElaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609
60. Rather S, Bala P (2019) Hybridization of constriction coefficient based particle swarm optimization and gravitational search algorithm for function optimization. In: International conference on advances in electronics, electrical, and computational intelligence (ICAEEC-2019)
61. Simon D (2008) Biogeography based optimization. IEEE Trans Evol Comput 12(6):702–713
62. Mirjalili S, Gandomi A, Mirjalili S, Saremi S, Faris H, Mirjalili S (2017) Salp swarm algorithm: a bioinspired optimizer for engineering design problems. Adv Eng Softw 854:1–29
63. Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl-Based Syst 96:120–133
64. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
65. Coello C (2000) Use of self-adaptive penalty approach for engineering optimization problems. Comput Ind 41(2):113–127
66. Dhiman G, Kumar V (2017) Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. Adv Eng Softw 114:48–70
67. Eskandar H, Sadollah A, Bahreininejad A, Hamdi M (2012) Water cycle algorithm-a novel metaheuristic optimization method for solving constrained engineering optimization problems. Comput Struct 110(111):151–166
68. Faramarzi A, Heidarinejad M, Stephens B, Mirjalili S (2019) Knowledge-based systems equilibrium optimizer: a novel optimization algorithm. Knowl Based Syst 191, Article ID 105190
69. Bayzidi H, Talatahari S, Saraee M, Lamarche CP (2021) Social network search for solving engineering optimization problems. Comput Intell Neurosci 85:2021
70. Sandgren E (1990) NIDP in mechanical design optimization. J Mech Des 112(2):223–229
71. Kaveh A, Dadras Eslamlou A (2020) Water strider algorithm: a new metaheuristic and applications. Structures 25:520–541
72. Kazemzadeh-Parsi MJ (2014) A modified firefly algorithm for engineering design optimization problems. Iranian Journal of Science and Technology. Trans Mech Eng 38(2):403
73. Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH (2020) Marine predators algorithm: a nature-inspired metaheuristic. Expert Syst Appl 152:113377
74. Siddall JN (1972) Analytical decision-making in engineering design. Prentice Hall, Hoboken
75. Ray T, Saini P (2001) Engineering design optimization using a swarm with an intelligent information sharing among individuals. Eng Optim 33(6):735–748
76. Han X, Yue L, Dong Y, Xu Q, Xie G, Xu X (2020) Efficient hybrid algorithm based on moth search and fireworks algorithm for solving numerical and constrained engineering optimization problems. J Supercomput 76:9404–9429
77. Chickermane H, Gea HC (1996) Structural optimization using a new local approximation method. Int J Numer Methods Eng 39(5):829–846
78. Rao SS (2009) Engineering optimization. Wiley, Hoboken
79. Parkinson A, Balling R, Hedengren JD (2018) Optimization methods for engineering design, 2nd edn. Brigham Young University, Brigham
80. Ravindran A, Ragsdell KM, Reklaitis GV (2006) Engineering optimization. Wiley, Hoboken
81. Amir HM, Hasegawa T (1989) Nonlinear mixed-discrete structural optimization. J Struct Eng 115(3):626–646

Authors and Affiliations

Absalom E. Ezugwu[1] · Jeffrey O. Agushaka[1,2] · Laith Abualigah[3,4,5] · Seyedali Mirjalili[6,7] · Amir H. Gandomi[8]

✉ Absalom E. Ezugwu
Ezugwua@ukzn.ac.za

✉ Laith Abualigah
laythdyabat@aau.edu.jo

Jeffrey O. Agushaka
jo.agushaka@science.fulafia.edu.ng

Seyedali Mirjalili
ali.mirjalili@torrens.edu.au

Amir H. Gandomi
gandomi@uts.edu.au

1 School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, King Edward Road, Pietermaritzburg 3201, KwaZulu-Natal, South Africa

2 Department of Computer Science, Federal University of Lafia, LafiaNasarawa State 950101, Nigeria

3 Faculty of Computer Sciences and Informatics, Amman Arab University, Amman 11953, Jordan

4 Faculty of Information Technology, Middle East University, Amman 11831, Jordan

5 School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang 11800, Malaysia

6 Centre for Artificial Intelligence Research and Optimization, Torrens University, Adelaide, Australia

7 Yonsei Frontier Lab, Yonsei University, Seoul, Korea

8 Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia