Knowledge-Based Systems 243 (2022) 108455

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A bottlenose dolphin optimizer: An application to solve dynamic emission economic dispatch problem in the microgrid

Abhishek Srivastava, Dushmanta Kumar Das *

Department of Electrical and Electronics Engineering, National Institute of Technology Nagaland, India

ARTICLE INFO

Article history:
Received 31 August 2021
Received in revised form 19 December 2021
Accepted 16 February 2022
Available online 22 February 2022

Keywords:
Mud ring feeding
Optimization techniques
Dynamic economic and emission dispatch (DEED) problem
Renewable energy sources (RES)

ABSTRACT

In this article, a new optimization technique is proposed. The proposed technique is a nature-inspired population-based meta-heuristic optimization technique which is inspired from bottlenose dolphins. Therefore, it is named as bottlenose dolphin optimizer. The proposed method replicates the mud ring feeding strategy used by them. In this study, mud ring feeding technique is mathematically modeled and is examined with two case studies for the superiority in comparison to other existing optimization techniques. The first case study includes benchmark functions which are used to examine the exploration, exploitation, local minima and convergence characteristics of the proposed technique. In second case study, a real-time optimization problem related to power systems i.e. a dynamic emission economic dispatch problem is considered. The two case studies show that the proposed technique exhibits a better exploration and exploitation capabilities as compared to some of the existing well known optimization algorithms.

1. Introduction

Searching for the best value is always a passion for human being. Nature and its elements are the sources of inspiration for human being to obtain the optimal value. Researchers are continuously giving their efforts to solve optimization problem from engineering, science, economics, etc.

In past decades, many optimization techniques have been developed and presented in various literature. These techniques are inspired from various sources. Based on the source of inspiration, these methods are classified into four main groups. The first group include evolution-based techniques which are inspired from laws of nature. This group include techniques such as genetic algorithm (GA) [1], evolution strategy (ES) [2], probability-based incremental learning (PBIL) [3], etc. The second group include techniques which replicate the physical laws of the universe. This group include algorithms such as gravitational search algorithm (GSA) [4], black hole (BH) [5] etc. Third group include techniques which are inspired from swarm behavior. Particle swarm optimization [6], whale optimization algorithm (WOA) [7], gray wolf optimization (GWO) algorithm [8], sailfish optimizer [9], harris hawks optimization (HHO) [10], salp swarm algorithm (SSA) [11] etc. are some of the popular and widely used techniques from this group. The last group include techniques inspired from human behavior. This group include techniques such as class topper optimization (CTO) [12], aggrandized CTO [13], teaching learning based optimization (TLBO) [14], league championship algorithm (LCA) [15], Kho-Kho optimization (KKO) algorithm [16], etc. Further, based on the population size, optimization techniques can be classified into two groups. This classification include population-based techniques like GWO, WOA, CTO etc. and single solution based techniques like tabu search technique [17].

In recent years, number of new optimization techniques have been developed and are applied to solve different optimization problems. Still, researchers make an attempt to solve these problems more efficiently by designing new and hybrid optimization schemes. But, the design of these schemes are of main concern. According to no free lunch (NFL) theory [18], none of these new techniques may be regarded as the universal best optimizer as they show equivalent performance when applied to all possible optimization problems. Further, [19] criticizes on the mathematical modeling of majority of the optimization algorithms. According to this study, superficial mathematical models with metaphor-based outfits should be avoided in order to make advancements in the field. Besides of these criticisms, designing an optimization scheme with excellent exploration, exploitation capabilities is changeling task. Things are more challenging when designed approaches should also maintain a balance between these phase. As a result, new optimization algorithms with specific goals of global and local searching strategies have emerged. These methods provide more options for researchers and experts

* Corresponding author.
E-mail addresses: abhishek91026@gmail.com (A. Srivastava), dushmantakumardas29@gmail.com (D.K. Das).

https://doi.org/10.1016/j.knosys.2022.108455
0950-7051/

in various fields. Hence, this paper also attempts to propose a new optimization technique, bottlenose dolphin optimizer which can be used to solve real-time optimization problems.

2. Contribution of this paper

In this paper, a new meta-heuristic optimization technique named as bottlenose dolphin optimizer (BDO) is proposed. The proposed BDO technique is inspired by the bottlenose dolphins and mimic the unique mud ring feeding technique used by them. As per the knowledge of authors, there is no existing study present on this domain. Still, it is worth mentioning here that [20] presents a dolphin swarm optimization (DSO) which is inspired by the biological characteristics and living habits of dolphin's. The DSO technique in [20] models habits of dolphins such as echolocation, co-operation and information exchange. Whereas, the proposed BDO technique presents the mathematical modeling of the mud ring feeding technique used by bottlenose dolphins.

Bottlenose dolphins are most interesting creatures living in the ocean. These dolphins feed on shrimps, squid, mollusks and cuttlefish. They use different hunting strategies among which the mud ring feeding is most interesting one. During this hunting process, a group of dolphins start searching for the prey location within the search space. Once the prey location is identified, a dolphin known as driver dolphin starts encircling the prey location. During encirclement, it bets the shore with its tail to create sand plumes. Plumes act as an artificial net around the prey. Slowly, dolphins move close towards the prey location and tightens the encirclement. As the encirclement gets smaller, fishes trapped inside, starts jumping out. Other members of the group hunt on the fishes jumping out of the encirclement.

In this study, this unique strategy is mathematically modeled. To show the superiority of the proposed optimization technique with respect to other existing techniques, two case studies are considered in this study. The first case study include benchmark functions used to examine the exploration, exploitation, convergence and local minima avoidance capabilities. In second case study, a real-time optimization problem related to power system i.e. dynamic economic load dispatch is solved using the proposed BDO technique.

3. Bottlenose dolphin optimizer

3.1. Inspiration

Bottlenose dolphins are the most interesting creatures living in the oceans except the region of Arctic and Antarctic circle. They gained popularity from the aquarium shows. These dolphins have a high encephalization level, third largest after humans and great apes, which increases their intelligence and emotional intelligence level [21]. Their intelligence not only helps them in object categorization, self-recognition, cultural knowledge transmission from generation to generation but also in their interaction with humans [22]. These dolphins can easily be trained. Therefore, they are used by Military people to locate sea mines and enemy divers in oceans. Also, the fishermen train them to driving fishes towards the hunting nets.

Apart from a good relation with humans, bottlenose dolphins have a good social interaction. They live in a fission–fusion society with a varying group size [23,24]. This means that one group can join another group or a group can also leave a existing group any time. Adult male dolphins generally live in pods for a short duration of time and mostly live alone or in a small group. Whereas, adult female dolphins and young dolphins always live in a group whose size can vary up to 15 dolphins [25]. Living in groups help dolphins in feeding on school of fishes which include shrimps, squid, mollusks and cuttlefish. Generally, they hunt in groups but can also hunt alone often on bottom-dwelling species. Group hunting help in maximizing the harvest of prey [25]. Dolphins can use different types of strategies like fish whacking [26], strand feeding [27], mud ring feeding [28] to hunt the prey.

Mud ring feeding also known as mud plume fishing, is a unique hunting strategy used by dolphins to forage and trap school of fishes. In the initial stage of hunting, dolphins living in a group cooperate with each other to search for a location of the prey. Once a school of fish is encountered, a single dolphin known as the driver dolphin starts encircling the fish group. During the encirclement process, they move their tail along the sand to create a plume. These plumes act like a temporary net around the fish which disorients the position of fish. Trapped fishes inside the plume try to move out of the plume by jumping out of it. Because of this jumping, other member of the group surround the location of the plumes and catch any fish coming to its location. To increase the efficiency of attack, dolphins reduce encirclement. Eventually, when dolphins are close to trap the fish location, more number of fishes jump out and are hunted by the dolphins. During this hunting process, to increase the efficiency of search, other dolphins of the group also create simultaneous plumes for hunting. This simultaneous plume increases the harvesting of preys for the dolphins.

This unique strategy used by bottlenose dolphins to forage and trap the prey inspires in the development of bottlenose dolphin optimizer (BDO) technique. The main objective for proposing BDO technique is to search for a global optimal solution from a randomly generated solutions effectively and more efficiently.

3.2. Mathematical modeling

The proposed BDO technique mimics the mud ring feeding behavior of Bottlenose dolphins. In the following section, mud ring feeding used by dolphins is mathematically modeled.

3.2.1. Initialization of population

Generally, Bottlenose dolphins live and hunt in groups. Therefore, proposed BDO technique is a population-based meta-heuristic optimization technique. In the initial stage, a random population of dolphins with their random position within the search space is initialized. Dolphins of this initialized population represents the candidate solution of the problem being solved. Whereas, their position represents the problem variables. Hence, the initialized population of dolphins with a population size of P is mathematically represented as follows.

$$D_{population} = \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_P \end{bmatrix}, \tag{1}$$

where D_1, D_2, ..., D_P saves the position of dolphins within search space.

3.2.2. Driver dolphin selection

Dolphins always cooperate with each other in order to search and trap the prey. They search for prey location and once the location of the prey is identified, a dolphin known as driver dolphin (P_{DD}), with best position among the others, start encircling the prey location. While the driver dolphin encircles the prey location, other member of the group follows it. The dolphins which follows the driver dolphins are referred as follower dolphins (P_{FD}). To select a driver dolphin from the initialized population, the fitness of each dolphin is evaluated. Fitness of each dolphin is

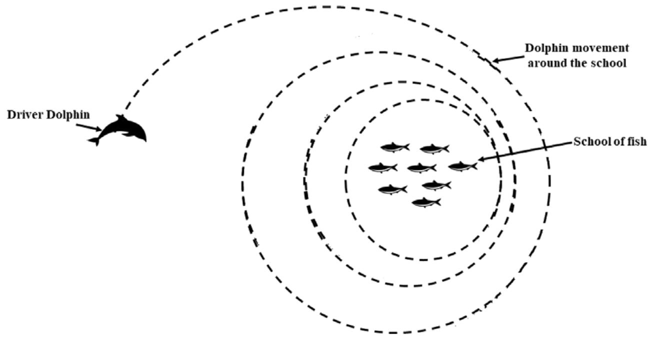


Fig. 1. Driver dolphin movement around the fish.

computed using the fitness function and the best dolphin (one with best solution) is referred as the driver dolphin while others are referred as follower dolphins. Mathematically, the selection of a driver dolphin is represented as follows.

$$D_{fitness} = \begin{bmatrix} D_1^f \\ D_2^f \\ \vdots \\ D_P^f \end{bmatrix} = \begin{bmatrix} f(D_1) \\ f(D_2) \\ \vdots \\ f(D_P) \end{bmatrix}, \tag{2}$$

where D_1^f, D_2^f, ..., D_1^p saves the fitness of the dolphins D_1, D_2, ..., D_P respectively, f represents the fitness function.

$$P_{DD}^n = best(D_{Fitness}) = f(D_q), \tag{3}$$

where P_{DD}^n saves the position of driver dolphin at a nth time instant, D_q represents the position of qth dolphin which exhibits the best fitness among the dolphin population.

3.2.3. Movement towards the prey

During the mud ring feeding process, dolphins switch between the searching stage and hunting stage depending upon their distance with respect to the prey location. With this switching strategy, the hunting efficiency of the dolphins is increased. In BDO technique, mud ring feeding strategy is also modeled in two stages (i) searching stage and (ii) hunting stage. The selection of these stages are done using a strategy randomizer factor S_r. S_r is evaluated depending on the ratio of present iteration i and total iteration (I_{max}) and is mathematically represented as follows.

$$S_r(i) = \frac{i}{I_{max}}. \tag{4}$$

If this ratio, $S_r(i) < S_r$ then dolphins use the searching strategy else if $S_r(i) > S_r$, they use hunting strategy. Mathematical modeling of these two strategy are discussed briefly in the following section. Depending upon a problem, the value of S_r can be tuned. Still, it is observed that when S_r is fixed to a value of 0.8, the proposed BDO technique provides effective results and converge to a global optimal solution.

Searching stage ($S_r(i) < S_r$):. During the searching stage, dolphins cooperate with each other and search for the location of the prey. Once the prey location is identified, driver dolphin (one with the best fitness) starts encircling the prey location. The movement of a driver dolphin towards the prey location is presented in Fig. 1. In BDO technique, it is assumed that the current position of a driver dolphin virtually represents the location of the prey. During the search process, this location is searched for a better solution. With reference to the encircling movement of dolphin and prey location as shown in Fig. 1, movement (position update) of a driver dolphin in the proposed BDO scheme is mathematically represented as follows.

$$P_{DD}^{i+1} = P_{DD}^i + P_{DD}^i \times rand \times e^{\theta} \times cos(2\pi \times \theta(i)), \tag{5}$$

where P_{DD}^{i+1} represents the updated position of the driver dolphin, P_{DD}^i is represents position of the driver dolphin, *rand* is a random number between [−1,1] which helps in diversing the searching ability, θ is a constant which is randomly decreased during the search and helps in encircling the location.

All other members of the group, i.e. the follower dolphins, follow the driver dolphin towards the prey location. But, sometimes some of the follower dolphins also start encircling other prey location. Probability of following driver dolphin or encircling other prey location is assumed to be 50%. In BDO technique, this is taken into consideration and a probability function p is assumed. p can vary between 0 to 1 and if $p < 0.5$, follower dolphins follows the driver dolphins, else if $p > 0.5$, follower dolphin encircles other prey location. Mathematically, movement of follower dolphins which follows the driver dolphin can be modeled as:

$$P_{FD}^{i+1} = P_{FD}^i + a_f \times rand \times (P_{DD}^i - P_{FD}^i), \tag{6}$$

where P_{FD}^{i+1} represents the updated position of a follower dolphin, P_{FD}^i represents the present position of the follower dolphin, P_{DD}^i represents the position of the driver dolphin, a_f is an acceleration factor which accelerates the movement of follower dolphin towards the driver dolphin. The acceleration factor can be tuned depending upon a problem. Still, it is observed that with a value of a_f between 2 to 4, the proposed BDO technique works effectively.

Further, the follower dolphins which encircles other prey location follow the encirclement strategy which is similar to driver dolphin. Under this situation, current position of the follower dolphin becomes the prey location and this location is searched for a better optimal solution. Movement of the follower dolphin which encircle other prey location is mathematically represented as follows.

$$P_{FD}^{i+1} = P_{FD}^i + P_{FD}^i \times rand \times e^{\theta} \times cos(2\pi \times \theta(i)), \tag{7}$$

where P_{FD}^{i+1} represents the updated position of the follower dolphin, P_{FD}^i is represents position of the follower dolphin,

Hunting stage ($S_r(i) > S_r$):. As the dolphins approaches closer to the prey location (optimal solution), the searching strategy is replaced with hunting strategy. During the process of hunting, the driver dolphin moves around the prey location in a circular path and reduces it distance towards the prey. This tightens the encirclement and this disturbs the prey i.e. fishes. Because of this, fishes start jumping out of the encirclement as presented in Fig. 2. This motion around the prey location increases the accuracy of hunting. In BDO technique, this process is mathematically modeled as follows.

$$P_{DD}^{i+1} = P_{DD}^i + P_{DD}^i \times rand \times sin(2\pi \times \theta(i)), \tag{8}$$

where P_{DD}^{i+1} is the updated position of driver dolphin, P_{DD}^i is the present position of the driver dolphin, *rand* is a random number [−1,1] which helps in enhancing the search ability, θ is a constant which is computed as presented in Section III-B4.

Similarly, to increase the efficiency of hunting, the follower dolphins takes position around the driver dolphin. This helps them to hunt the fishes jumping out of the encirclement. In BDO technique, this process of occupying position around the driver dolphin location is graphically presented in Fig. 3 and is mathematically modeled as follows.

$$P_{FD}^{i+1}(v) = P_{DD}^i(v) + rand \times 2\pi \times P_p \times r_p, \tag{9}$$

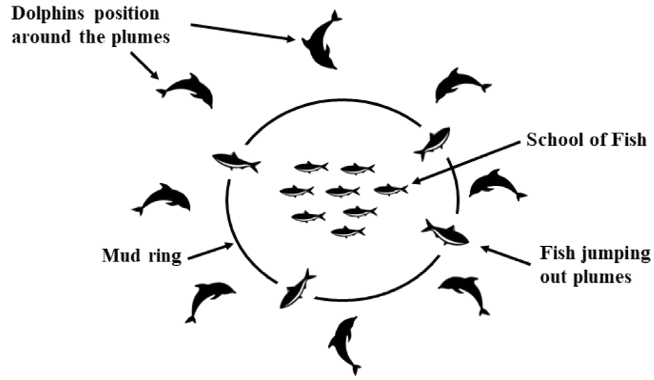


Fig. 2. Dolphins attacking strategy.

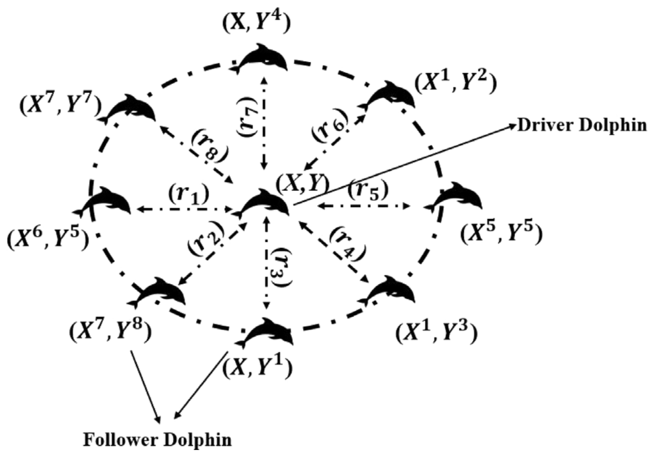


Fig. 3. Hunting strategy.

where P_{FD}^{i+1} is the updated position of follower dolphin, P_{DD}^{i} is the present position of the driver dolphin, r_p is $P_{DD}^{i}(v) - P_{FD}^{i}(v)$ and P_p is computed according to the fitness of the dolphins in the group and is computed as:

$$Pp = \frac{r_i}{T}, \tag{10}$$

where r_i is the rank of ith dolphin, T is the population of dolphins excluding the driver dolphin.

3.2.4. Calculation of θ

θ plays an important role in the search of prey (global best) location. θ is linearly decreased during the searching process to confine the search space towards the prey location (global optimal location). During the searching stage, θ is linearly decreased from 1 to θ_{min} using

$$\theta(i) = 1 - (1 - \theta_{min}) \times \frac{i}{i_{max}}, \tag{11}$$

where i is the present iteration, i_{max} is the total iterations.

By linearly decreasing θ, the function $cos(2\pi \times \theta)$ is gradually decreased which helps in confining the search. During the hunting stage, to increase the efficiency of obtaining the global optimal solution, θ is decreased from θ_{min} to 0 using

$$\theta(i) = \theta_{min} - \theta_{min} \times \left(\frac{i}{i_{max}}\right)^2, \tag{12}$$

where i is the present iteration, i_{max} is the total iterations.

With the help of (11) and (12), a time varying variable θ is generated whose value is decreased from 1 to 0 as i is varied from

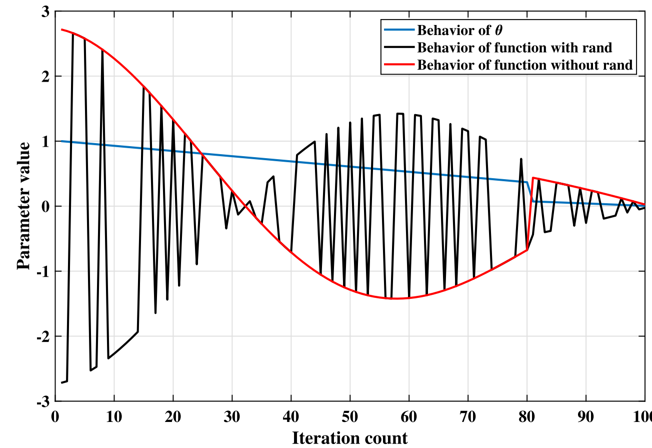


Fig. 4. Behavior of parameter (θ).

0 to I_{max}. This helps the BDO method to preform exploration and exploitation by varying the functions $rand \times e^{\theta(i)} \times cos(2\pi\theta(i))$ from (5) and $sin(2\pi\theta(i))$ from (8) as θ varies from 1 to 0. Behavior of these function with and without *rand* along with θ is presented in Fig. 4. It is observed from Fig. 4 that function with *rand* provides better diversity in the search in comparison to the function without *rand*. From Fig. 4, the behavior of θ can be observed as it varies from 1 to 0. It is worth mentioning here that why a cosine function is used in search stage and sine function during hunting stage. In the initial stages, θ is one, hence cosine function provides a good searching abilities as its value is close to one which is opposite in case of sine whose value is close to zero. In later stages, i.e. hunting stage, the value of θ is close to zero, sine function provides value close to zero which helps in fine tuning. If in this stage, a cosine function is used whose value is almost equal to one, then the objective of fine tuning cannot be achieved.

In order to further explain the flow of the proposed BDO scheme, pseudo code and flowcharts of the alogorithm are presneted in Figs. 5 and 6 respectively.

3.3. Complexity analysis of BDO

To analyze the run-time complexity of the proposed BDO technique, in this section an asymptotic analysis of the proposed method is carried out. According to [16], the size of input parameters can directly influence the run-time performance of an optimization algorithm. Therefore, to compute the run-time performance of the proposed BDO technique, all the input parameters are considered in this study. The input parameters in BDO technique include parameters like number of iterations (I_{max}), size of dolphins population (D_p) and size (dimension) of an optimization problem (OP_{max}). On the basis of these input parameters, the operations in BDO technique to be executed are as follows:

1. $N_{max} + 1$
2. $N_{max} \times (D_p + 1)$
3. N_{max}
4. $N_{max} \times D_p \times OP_{max}$
5. $N_{max} \times D_p \times OP_{max}$
6. N_{max}
7. $N_{max} \times D_p \times OP_{max}$
8. $N_{max} \times (D_p + 1)$

Depending upon the above mentioned operations, the total number of operations (TO_{total}) can be computed as:

$$TO_{total} = (N_{max} + 1) + (N_{max} \times (D_p + 1))$$

Initialize the population of dolphins randomly
Initialize parameters $a_f, N_{max}, \theta_{min1}, S_r$
While the termination condition is not satisfied
    for each dolphin
        Compute the fitness of each dolphins using Eq. (2)
        Selected the fittest dolphin assume it as the driver dolphin using Eq. (3)
    end for
    Update $S_r(n)$ using Eq. (4)
    if $S_r(n) < S_r$
        Update $\theta(n)$ using Eq. (11) and p
        Update the position of Driver dolphin using Eq. (5)
        if $p < 0.5$
            Update the position of Follower dolphins using Eq. (6)
        else if $p > 0.5$
            Update the position of Follower dolphins using Eq. (7)
        end if
    end if
    if $S_r(n) > S_r$
        Update $\theta(n)$ using Eq. (12)
        Update position of Driver dolphin using Eq. (8)
        Update the position of Follower dolphins using Eq. (9)
    end if
    Check if all solutions are with the search range and amend it
end while
Return Driver dolphin

Fig. 5. Pseudo code for BDO technique.

$$\begin{aligned} &+(N_{max}) + (N_{max} \times D_p \times OP_{max}) \\ &+(N_{max} \times D_p \times OP_{max}) + (N_{max}) \\ &+(N_{max} \times D_p \times OP_{max}) \\ &+N_{max} \times (D_p + 1), \end{aligned} \tag{13}$$

$$\begin{aligned} O_{total} = &\ 3 \times (N_{max} \times D_p \times OP_{max}) \\ &+2 \times (N_{max} \times D_p) \\ &+5 \times (N_{max}) + 1. \end{aligned} \tag{14}$$

To analyze the run-time performance, now, all the input parameters are assumed to be equal for a worst case scenario. Under this assumption, (14) can be written as a function of n and this function $f(n)$ thus is represented as:

$$f(n) = 3 \times n^3 + 2 \times n^2 + 5 \times n + 1 \tag{15}$$

Here, it is now assumed that $f(n)$ is big-O of n^3, i.e. $g(n) = n^3$. Therefore, the time complexity of the proposed BDO technique is given as $O(n^3)$.

4. Case study A: Benchmark functions

Many optimization techniques have been presented in various literatures to solve real-time optimization problems. But, many of these techniques fail to perform effectively under large and complex problem scenario. To solve large and complex problems, it is important that an optimization technique exhibits a good exploration, exploitation, local minima avoidance and convergence capabilities. To evaluate these capabilities of the proposed BDO technique, two suites of benchmark functions are considered in this study. The first suite comprises of twenty-nine functions including seven uni-modal functions (F_1 to F_7), sixteen multi-modal functions (F_8 to F_{23}) and six composite functions (F_{24} to F_{29}) which are taken from [8]. Whereas, the second suite include ten benchmark functions from the CEC-2020 test bed. All of these functions are tested using the LabVIEW©2015 simulator, which was installed on a 64 bit computer with an Intel (R) core (TM) i7 processor running at 2.0 GHz. This study's details and key findings are presented below.

4.1. Suite A: Classical and CEC-2005 composite functions

To test the capabilities of the proposed BDO strategy to solve the benchmark functions of test suit A, each function is simulated thirty times. The dimension for these function are considered to be 30. Whereas, parameters of BDO scheme set to solve these functions are discussed in Table 1. After thirty simulations, the results obtained are used to compute statistical results such as average value, standard deviation, and, average computation time. These obtained results are seen in Table 2. Further, to show the supremacy of the proposed scheme over some existing techniques such as GSA [4], DE [29], PSO [30], GA [10], GWO [8], HHO [10], CTO [12], A-CTO [13], and KKO [16], a comparative study using the existing results is provided. The results for GSA, DE, PSO, GA, HHO, CTO, A-CTO, and KKO, are taken same as reported in the works [8,10,13,16] and these results are also reported in Table 2. For a fair comparison between the different optimizers and the proposed BDO scheme, two important parameters, the population size and maximum iteration count are set to 30 and 500 respectively.

From the results reported in Table 1, by looking at the average value, it is clear that the proposed BDO scheme outperforms the

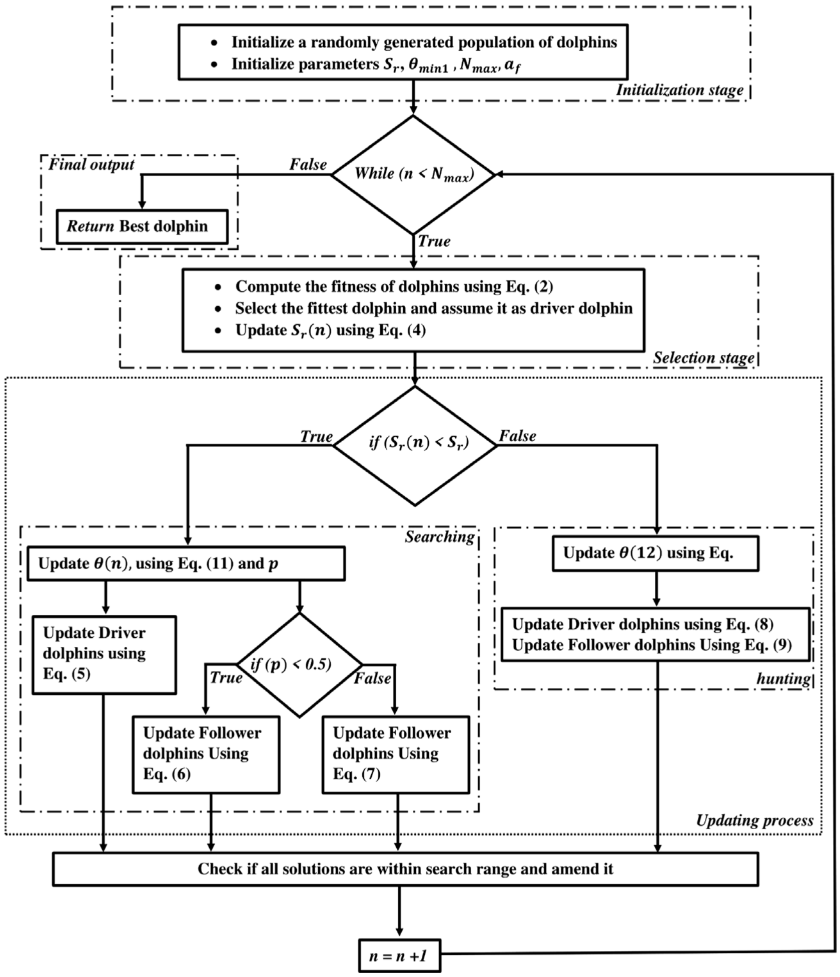


Fig. 6. Flow chart for BDO technique.

Table 1
Parameter setting for different optimizers to solve benchmark functions.

Method	Parameter setting
DE	Suite A : Result taken from [8] Suite B : –
GA	Suite A : Result taken from [10] Suite B : –
HHO	Suite A : Result taken from [10] Suite B : –
PSO	Suite A : Result taken from [8] Suite B : Inertia factor - 0.5; $c_1 = c_2 = 2$
GWO	Suite A : Result taken from [8] Suite B : Convergence constant = [0, 2]
CTO	Suite A : Result taken from [13] Suite B : Section = 3; c = 1.5; weight max = 0.5; weight min = 0
A-CTO	Suite A : Result taken from [13] Suite B : Section = 3; c = 1.5; weight max = 4 0.7; weight min = 5, crossover probability (C_p) = 0.7, weight coefficient (μ) = 1.5
KKO	Suite A : Result taken from [16] Suite B : Endurance factor e = [2, 0]
BDO	Suite A : $a_f = 3.5$, $\theta_{min} = 0.4$; $S_r = 0.8$ Suite B : $a_f = 3.5$, $\theta_{min} = 0.4$; $S_r = 0.8$

other methods in sixteen of the twenty-nine functions tested. Based on the findings of this analysis, it is possible to conclude that the proposed BDO scheme has a higher precision than the techniques used for comparison. Furthermore, it is discovered that the proposed BDO scheme outperforms in twenty cases by analyzing the standard deviation, showing the robustness of proposed algorithm. According to this report, it may be concluded that the proposed BDO scheme has exceptional exploration and exploitation capability. Furthermore, a proper balance is maintained between exploration and exploitation, assisting in the prevention of local minima. From the results presented in Table 1, it is also found that the computational time of BDO to solve these functions is found to be shorter than that of CTO and A-CTO. Whereas, it is more in comparison to KKO. To analyze the rate of convergence of the proposed BDO technique, convergence behavior of F_3, F_{10}, F_{18} and F_{25} are examined. For a comparative analysis, convergence behavior for existing techniques such as PSO, GWO, CTO, A-CTO and KKO are also examined for these function. The convergence plots are presented in Fig. 7 for these functions. It is clear from these figures that the proposed BDO scheme exhibits a faster convergence profile. Apart from this result for the non-parametric analysis of the proposed method, Table 3 provides p-value with a significance of 5% for t-test which is evaluated to check whether there is a significant difference between the results obtained using BDO and other popular techniques like PSO, CTO, A-CTO, and KKO. This results show that BDO scheme shows significant difference between the results obtained using other schemes used for comparison.

Table 2
Comparative analysis of average value and standard deviation for benchmark functions.

OF's	Parameter	GWO [8]	PSO [8]	DE [8]	GA [10]	HHO [10]	CTO [12]	A-CTO [13]	KKO [16]	BDO
F_1	Average	6.59E−28	0.000136	8.20E−14	1.03E+03	3.95E−97	3.50E−19	1.10E−20	1.30E−35	2.26E−128
	Deviation	6.34E−05	0.000202	5.90E−14	5.79E+02	1.72E−96	6.59E−19	4.52E−21	3.74E−35	7.14E−124
	Time (s)	–	–	–	–	–	20.396	19.308	0.221	9.67
F_2	Average	7.18E−17	0.042144	1.50E−09	2.47E+01	1.56E−51	5.44E−10	5.63E−14	3.60E−23	3.93E−49
	Deviation	0.029014	0.045421	9.90E−10	5.68E+00	6.98E−51	5.96E−10	7.73E−14	6.41E−23	7.14E−124
	Time (s)	–	–	–	–	–	23.103	19.681	0.226	9.679
F_3	Average	3.29E−06	70.12562	6.80E−11	2.65E+04	1.92E−63	1.12E−17	6.34E−18	1.58E−35	7.21E−121
	Deviation	79.14958	22.11924	7.40E−11	3.44E+03	1.05E−62	2.96E−17	8.53E−18	5.48E−35	2.19E−120
	Time (s)	–	–	–	–	–	20.739	19.764	0.23	9.818
F_4	Average	5.61E−07	1.086481	0	5.17E+01	1.02E−47	4.52507	1.26E+01	6.05E−19	1.38E−61
	Deviation	1.315088	0.317039	0	1.05E+01	5.01E−47	1.09056	1.64	2.01E−18	2.83E−61
	Time (s)	–	–	–	–	–	21.3	20.056	0.232	9.855
F_5	Average	26.81258	96.71832	0	1.95E+04	1.32E−02	1.30E−19	26.1	28.90336	2.69E+01
	Deviation	69.90499	60.11559	0	1.31E+04	1.87E−02	2.11E−19	2.68	0.038939	2.07E−01
	Time (s)	–	–	–	–	–	22.85	20.086	0.256	9.886
F_6	Average	0.816579	0.000102	0	9.01E+02	1.15E−04	6.12E−08	4.69E−20	5.788447	1.6144
	Deviation	0.000126	8.28E−05	0	2.84E+02	1.56E−04	8.64E−08	7.82E−20	0.911463	1.50E−01
	Time (s)	–	–	–	–	–	22.338	20.562	0.265	9.917
F_7	Average	0.002213	0.122854	0.00463	1.91E−01	1.40E−04	0.005434	6.90E−02	8.77E−05	3.07E−05
	Deviation	0.100286	0.044957	0.0012	1.50E−01	1.07E−04	0.002232	1.51E−02	8.29E−05	2.37E−05
	Time (s)	–	–	–	–	–	21.841	20.94	0.278	10.722
F_8	Average	−6123.1	−4841.29	−11 080.1	−1.26E+04	−1.25E+04	−2791.7	−5408.7	−3758.63	−6800.39
	Deviation	−4087.44	1152.814	574.7	4.51E+00	1.47E+02	330.711	146.75	408.5173	7.39E+02
	Time (s)	–	–	–	–	–	282.838	244.457	75.671	86.719
F_9	Average	0.310521	46.70423	69.2	9.04E+00	0.00E+00	799.699	236.432	0	0.00E+00
	Deviation	47.35612	11.62938	38.8	4.58E+00	0.00E+00	0.651061	0.02842	0	0
	Time (s)	–	–	–	–	–	280.269	244.857	76.723	87.089
F_{10}	Average	1.06E−13	0.276015	9.70E−08	1.36E+01	8.88E−16	1.5035	1.55	11.24042	0
	Deviation	0.077835	0.50901	4.20E−08	1.51E+00	4.01E−31	0.651061	2.20E−01	9.234612	0
	Time (s)	–	–	–	–	–	260.443	246.261	76.948	88.148
F_{11}	Average	0.004485	0.009215	0	1.01E+01	0.00E+00	0.211515	0.012671	0	0
	Deviation	0.006659	0.007724	0	2.43E+00	0.00E+00	0.018247	0.005913	0	0
	Time (s)	–	–	–	–	–	266.348	246.693	77.793	89.226
F_{12}	Average	0.53438	0.006917	7.90E−15	4.77E+00	2.08E−06	6.08416	2.178678	−0.30359	−0.90558
	Deviation	0.020734	0.026301	8.00E−15	1.56E+00	1.19E−05	4.80515	2.792436	0.360173	0.076908
	Time (s)	–	–	–	–	–	260.411	286.418	82.782	89.582
F_{13}	Average	0.654464	0.00675	5.10E−14	1.52E+01	1.57E−04	4.065	3.35046	3.302702	1.186892
	Deviation	0.004474	0.008907	4.80E−14	4.52E+00	2.15E−04	3.1734	4.6439	0.729056	0.22259
	Time (s)	–	–	–	–	–	282.699	298.157	88.844	92.819
F_{14}	Average	4.042493	3.627168	0.998004	9.98E−01	9.98E−01	0.998004	0.998004	1.28144	0.998004
	Deviation	4.252799	2.560828	3.30E−16	4.52E−16	9.23E−01	1.22E−16	0	0.478666	0
	Time (s)	–	–	–	–	–	43.56	41.218	13.505	14.282
F_{15}	Average	0.000337	0.000577	4.50E−14	3.33E−02	3.10E−04	0.000434	0.000469	0.002303	0.000553
	Deviation	0.000625	0.000222	0.00033	2.70E−02	1.97E−04	0.000238	0.000274	0.001076	0.00258
	Time (s)	–	–	–	–	–	44.663	41.967	13.524	14.412
F_{16}	Average	−1.03163	−1.03163	−1.03163	−3.78E−01	−1.03E+00	1.03136	−1.03136	−1.02271	−1.03136
	Deviation	−1.03163	6.25E−16	3.10E−13	3.42E−01	6.78E−16	0	0	0.01186	0
	Time (s)	–	–	–	–	–	43.713	42.081	13.62	14.469
F_{17}	Average	0.397889	0.397887	0.397887	5.24E−01	3.98E−01	0.397887	0.397887	0.412356	0.397887
	Deviation	0.397887	0	9.40E−09	6.06E−02	2.54E−06	0	0	0.017307	0
	Time (s)	–	–	–	–	–	45.84	42.427	13.744	14.835
F_{18}	Average	3.000028	3	3	3.00E+00	3.00E+00	3	3	3	3
	Deviation	3	1.33E−15	2.00E−15	0.00E+00	0.00E+00	0	0	0	0
	Time (s)	–	–	–	–	–	44.999	42.913	13.746	14.878
F_{19}	Average	−3.86263	−3.86278	NA	−3.42E+00	−3.86E+00	−3.87943	−3.87943	−3.59924	−3.87931
	Deviation	−3.86278	2.58E−15	NA	3.03E−01	2.44E−03	2.56E−16	0	0.12549	0.000318
	Time (s)	–	–	–	–	–	42.321	43.227	13.817	14.91
F_{20}	Average	−3.28654	−3.26634	NA	−1.61351	−3.322	−3.3222	−3.23073	−2.13964	−3.26061
	Deviation	−3.25056	0.060516	NA	0.46049	0.137406	0	0.0518	0.31007	0.064928
	Time (s)	–	–	–	–	–	46.854	44.38	13.817	15.222
F_{21}	Average	−10.1514	−6.8651	−10.1532	−6.66177	−10.1451	−3.86903	−6.38539	−2.90419	−6.9779
	Deviation	−9.19015	3.019644	2.5E−06	3.732521	0.885673	1.36967	3.065909	1.482463	2.504847
	Time (s)	–	–	–	–	–	43.613	44.656	13.887	15.603
F_{22}	Average	−10.4015	−8.45653	−10.4029	−5.58399	−10.4015	−10.4029	−9.2898	−4.14019	−10.3175
	Deviation	−8.58441	3.087094	3.90E−07	2.605837	1.352375	1.26E−15	2.726527	−1.09318	0.211289
	Time (s)	–	–	–	–	–	47.066	45.124	13.899	15.692

(*continued on next page*)

Table 2 (*continued*).

OF's	Parameter	GWO [8]	PSO [8]	DE [8]	GA [10]	HHO [10]	CTO [12]	A-CTO [13]	KKO [16]	BDO
F_{23}	Average	−10.5343	−9.95291	−10.5364	−4.69882	−10.5364	−10.5364	−10.5364	−3.97328	−8.65731
	Deviation	−8.55899	1.782786	1.90E−07	3.256702	0.927655	1.26E−15	0	0.674575	2.18054
	Time (s)	–	–	–	–	–	43.983	45.959	14.138	16.123
F_{24}	Average	43.83544	100	6.75E−02	626.8389	396.8256	135.843	152.843	3.94335	2.11
	Deviation	69.86146	81.65	1.11E−01	101.2255	79.58214	54.94	49.78	8.39575	4.46
	Time (s)	–	–	–	–	–	61.5978	59.2569	25.9562	28.6951
F_{25}	Average	91.80086	155.91	28.759	999.4998	910	120.164	100.164	4.4041	1.22E−08
	Deviation	95.5518	13.176	8.6277	29.44366	0	40.952	21.754	8.91397	1.97E−08
	Time (s)	–	–	–	–	–	61.9897	60.2659	25.6259	2.90E+01
F_{26}	Average	61.43776	172.03	144.41	998.9091	910	102.949	84.949	6.3271	1.21E−09
	Deviation	68.68816	32.769	19.401	25.27817	0	15.18	32.41	23.8971	3.12E−09
	Time (s)	–	–	–	–	–	62.5897	61.6929	26.5652	2.95E+01
F_{27}	Average	123.1235	314.3	324.86	1002.032	910	165.149	123.149	8.76504	2.672925
	Deviation	163.9937	20.066	14.784	26.66321	0	64.567	97.87	30.4813	4.395
	Time (s)	–	–	–	–	–	63.1597	63.4592	26.6652	29.8956
F_{28}	Average	102.1429	83.45	10.789	1512.467	860.8925	197.431	154.521	9.8846	2.06E−11
	Deviation	81.25536	101.11	2.604	94.64553	0.651222	15.169	5.918	36.2415	3.79E−11
	Time (s)	–	–	–	–	–	63.891	62.6691	26.4912	3.13E+01
F_{29}	Average	43.14261	861.42	490.94	1937.396	558.9653	146.745	135.46	4.13161	1
	Deviation	84.48573	125.81	39.461	1.13E+01	5.11E+00	34.491	12.686	36.2415	3.16
	Time (s)	–	–	–	–	–	65.2691	66.5591	26.5959	32.5981

Table 3
Non-parametric statistical analysis of BDO with respect to PSO, CTO, A-CTO and KKO.

OF's	PSO [30]	CTO [12]	A-CTO [13]	KKO [16]	OF's	PSO [30]	CTO [12]	A-CTO [13]	KKO [16]
F_1	6.78E−03	0.03385	1.96E−02	2.18E−02	F_{16}	2.33E−02	0.008734	4.27E−02	1.94E−03
F_2	7.61E−06	0.013933	8.05E−03	1.74E−02	F_{17}	1.52E−02	0.00544	4.71E−02	2.59E−02
F_3	0.00085	0.013824	0.0491407	0.0246959	F_{18}	0.032613	0.01321	0.0458197	0.001346
F_4	2.78E−10	0.025074	3.81E−02	1.33E−03	F_{19}	5.69E−03	0.015337	5.91E−03	2.74E−02
F_5	1.29E−02	0.023393	1.81E−02	3.10E−02	F_{20}	3.57E−01	0.042717	2.61E−02	2.75E−02
F_6	5.78E−16	0.009329	1.97E−02	0.0317794	F_{21}	0.000572	0.045705	0.00862421	0.041203
F_7	2.99E−05	0.028651	2.58E−02	2.71E−02	F_{22}	3.94E−02	0.004556	2.57E−02	4.36E−02
F_8	4.75E−03	0.006445	1.18E−02	4.99E−02	F_{23}	1.05E−02	0.045784	2.83E−02	8.72E−03
F_9	6.28E−06	0.041982	0.0179351	0.000965865	F_{24}	0.008576	0.048766	0.0481772	0.005199
F_{10}	8.86E−03	0.038421	4.01E−02	4.92E−02	F_{25}	1.39E−03	0.009115	4.87E−02	2.81E−02
F_{11}	2.50E−02	0.013577	9.06E−03	1.72E−02	F_{26}	4.18E−02	0.036511	1.96E−02	1.08E−02
F_{12}	0.00651	0.033879	6.53E−03	0.0131217	F_{27}	1.63E−13	0.024958	1.20E−02	0.022317
F_{13}	5.01E−04	0.025658	1.41E−02	1.80E−02	F_{28}	9.82E−38	0.041311	1.79E−02	4.74E−02
F_{14}	8.82E−02	0.042558	5.31E−03	3.80E−02	F_{29}	9.80E−11	0.02734	4.64E−02	1.28E−02
F_{15}	0.0135	0.000396	0.0258014	0.0292063					

4.2. Suite B: CEC-2020 benchmark function

To further explore the exploration, exploitation, local minima avoidance and convergence capabilities of the proposed BDO technique, CEC-2020 benchmark functions are considered in this section. CEC-2020 functions include a mixture of uni-modal, multi-modal, hybrid, and composite functions which are ideal for the validation of above mentioned capabilities of an optimization scheme. To evaluate the performance, each of these functions are tested with a small problem dimension considering 30 problem variables (30D) and high problem dimension considering 500 problem variables (500D). Each function is simulated for fifteen times in LabVIEW platform. The parameters for BDO set to solve these functions are reported in Table 1. The results obtained for these fifteen runs are used to compute statistical parameters like average value, standard deviation, average computation time. These results are presented in Table 4 considering 30 dimensional problem and Table 5 considering 500 dimensional problem. The obtained results are compared with some of the exiting optimization scheme such as PSO [31], GWO [8], CTO [12], A-CTO [13], and KKO [16] to show the superiority of the proposed BDO technique. The parameter setting for these optimizer are discussed in Table 1. Whereas the results obtained are presented in Tables 4 and 5. It is to be noted here that for a fair comparison each optimizer is simulated with a population size of 50 and iteration count of 500 and 20 000 while considering 30D and 500D problems respectively.

Based on the results presented in Tables 4 and 5, it is evident that the proposed BDO scheme shows a better exploration and exploitation capability in comparison to some of the other optimization schemes used for the analysis. The suggested BDO scheme achieves the best mean outcome for most of the functions under both small and high dimension problems. This demonstrates that BDO can solve these functions effectively. Examining the deviation, the outcomes show that the proposed BDO scheme is more stable and robust than the other schemes under consideration, as the results deviates less in seven and six of ten cases while considering small and high problem dimensions. The computation burden on BDO is found less as compared to some of the other optimizers used for comparison. To assess the convergence behavior of the proposed BDO strategy, the convergence plots for some random functions such as F_1, F_4, F_7 and F_9 are presented in Fig. 8 for 30 dimensional problem and Fig. 9 for 500 dimensional problem. In the majority of scenarios, the plots reveal that BDO scheme has a higher convergence rate than PSO, GWO CTO, A-CTO, and KKO. Next, the non-parametric analysis considering p-value with a significance of 5% for t-test is conducted on the results obtained using BDO and other existing techniques like PSO, GWO CTO, A-CTO, and KKO. These results are presented in Tables 4 and 5.

The results obtained using the proposed BDO technique for the benchmark functions demonstrate that the BDO methodology has strong exploration and exploitation capability, which may aid in the solving of complex optimization problems.

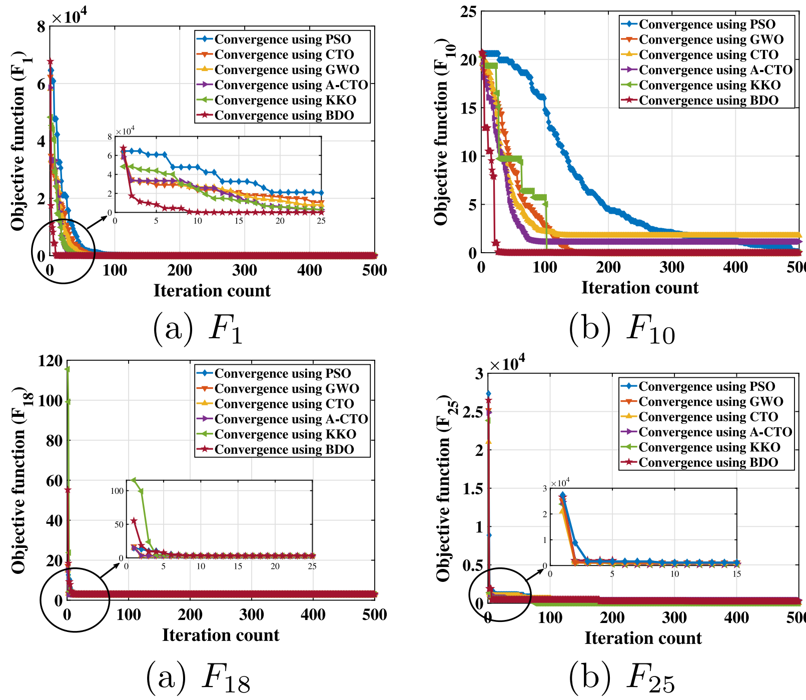


Fig. 7. Convergence of objective function for functions F_3, F_{10}, F_{18} and F_{25}.

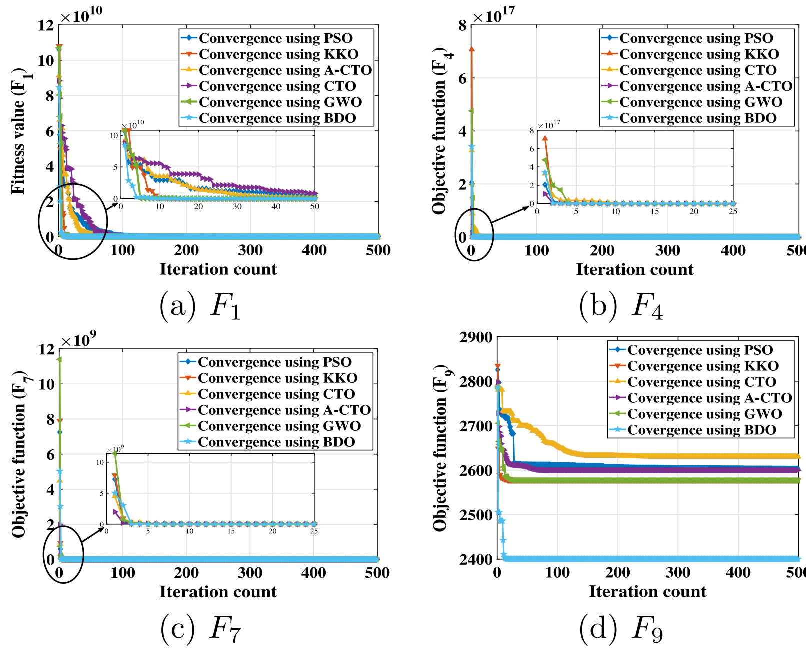
Table 4
Statistical results for benchmark function using different optimizer considering 30*D* problem.

Method	Parameter	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
PSO [31]	Average	641.509	1234.15	815.105	1.52E+12	1716.65	1908.88	2124.48	2300.06	2602.34	2903.05
	Deviation	431.694	184.074	28.6848	1.16E+12	1.38536	476.37	27.3552	0.103923	1.01681	0.610246
	p-Value	0.000127	0.006782	1.46E−03	8.47E−05	4.72E−03	0.013629	0.009455	0.020914	4.23E−04	1.30E−03
	Time (s)	36.206	36.737	36.094	36.892	37.165	36.666	36.405	62.276	62.652	63.204
GWO [8]	Average	100	1118.947	999.197	1900.88	1700	1624.173	2131.123	2751.857	2726	2838.167
	Deviation	0	32.81659	36.73451	0	0	27.26826	39.38396	648.8724	129.7356	138.7137
	p-Value	0.045669	0.036156	0.048169	0.029813	0.00044	0.034219	0.036733	0.049675	0.023083	0.012882
	Time (s)	23.04	23.914	23.404	25.632	21.601	25.722	29.447	64.529	66.126	69.011
CTO [12]	Average	1.94E+06	7.43E+03	9.37E+02	4.16E+03	3.22E+03	2.55E+03	2.96E+03	2.70E+03	2.62E+03	2.91E+03
	Deviation	1 756 300	708.4336	30.40482	720.9728	93.68339	107.6689	493.0051	134.7701	8.646903	0.281603
	p-Value	0.003095	0.016851	0.046336	0.032876	0.011419	0.041355	0.005839	0.002622	0.032476	0.017897
	Time (s)	111.862	113.561	114.01	110.431	114.807	117.003	114.459	250.499	255.591	260.592
A-CTO [13]	Average	100.3513	2734.867	803.6017	2179.863	4380.063	2267.447	3566.887	2300.047	2600.033	2906.673
	Deviation	0.461257	304.8317	22.74366	471.9509	1058.436	119.9204	258.2601	0.040415	0.057735	5.066846
	p-Value	0.00271	0.029778	0.029509	0.028615	0.01801	0.030895	0.042277	0.031397	0.023071	0.045501
	Time (s)	106.39	111.522	112.51	111.157	113.603	111.602	113.109	251.608	254.757	258.89
KKO [16]	Average	100	1100	1034.32	1900.89	1700	1608.477	2108.47	2377.34	2577.313	2678
	Deviation	0	0	16.27517	0	0	0.025166	0.01	0.01	0.005774	0
	p-Value	0.03116	0.028365	0.037921	0.001953	0.03628	0.040103	0.049558	0.00779	0.019237	0.014983
	Time (s)	22.6	27.984	21.264	28.47	24.798	22.049	22.434	52.527	52.76	54.001
BDO	Average	100	1100	921.8117	1900.753	1700	1605.717	2105.74	2266.667	2400	2600.887
	Deviation	0	0	35.14626	0.035119	0	0.023094	0	57.73503	0	0.060277
	Time (s)	20.459	22.849	24.41	20.485	29.526	25.533	29.867	65.753	66.169	67.269

5. Salient features of BDO technique

In recent years, many optimization techniques have been developed and used to solve many optimization problems. Still, researchers try to develop new methods to get a more reliable, stable and improved results for these problems and also solve other unsolved optimization problems. The proposed BDO method is one of such newly developed methods. The proposed BDO optimization technique exhibits some important features which not only help it in exploring the search space for a optimal solution at a faster rate but also differentiates it from other existing methods. This section presents a brief discussion on these features which differentiates BDO from other existing methods.

- **Grouping and searching Strategy:** Dolphins live in small groups. During the hunting process, a driver dolphin (closest to the prey location) starts encircling the prey location in order to trap the prey and find more optimal position to hunt them. While the other dolphins of the group either follows the driver dolphins (moves towards the best) or encircle other locations (search for other locations) in search of the prey location. In the proposed BDO technique this grouping and searching strategy is replicated. The driver dolphin is assumed to be the fittest search agent from the population. This solution exploits it own solution in order to search for a more optimal solution using (5). Whereas, the rest of the population can either follow the best solution



(a) F_1 (b) F_4 (c) F_7 (d) F_9

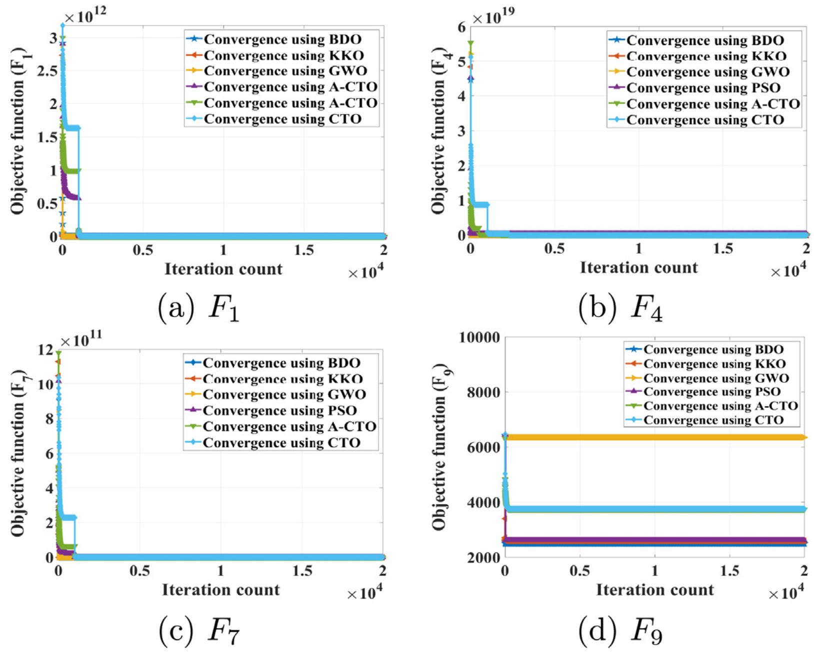
Fig. 8. Convergence behavior of objective function considering 30*D* problem.

Table 5
Statistical results for benchmark function using different optimizer considering 500*D* problem.

Method	Parameter	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
PSO [30]	Average	7.01E+06	35 788.3	7.34E+04	3.98E+06	77 470.7	4.66E+06	1.98E+06	28 403.9	2612.14	2912.67
	Deviation	43.1359	74.8417	65.191	9.13E+01	21.5602	9.63682	27.1679	13.9175	2.5035	2.74639
	p-Value	0.038912	0.039369	2.67E−03	8.46E−03	2.31E−02	0.014724	0.013009	0.032001	4.01E−02	4.92E−02
	Time (s)	1516.66	1538.9	1511.97	1545.4	1556.83	1535.93	1525	2608.72	2624.47	2647.6
GWO [8]	Average	100	1100.01	7697.69	1900.01	1700	1749.3	2249.39	218 260	6348.97	2916.39
	Deviation	0	0	0.091381	0	0.298033	0	0.069186	0.812041	0.13585	0.75628
	p-Value	0.0092	0.012796	0.04722	0.036991	0.017305	0.04569	0.049831	0.036332	0.031595	0.032624
	Time (s)	965.139	1001.75	980.387	1073.72	904.86	1077.49	1317.31	2703.1	2764.72	2890.85
CTO [12]	Average	1.63E+08	134 599	6.02E+04	8.67E+06	4.10E+06	1.45E+06	2.29E+06	139 327	3754.11	2911.94
	Deviation	0.677376	0.667935	0.11215	0.107733	0.271381	0.226044	0.516993	0.905976	0.507463	0.392811
	p-Value	0.042808	0.017606	0.040467	0.011392	0.001732	0.029238	0.016421	0.040385	0.040342	0.016315
	Time (s)	4685.87	4757.04	4775.85	4625.92	4809.23	4901.22	4794.66	10 493.3	107 06.6	10 916.1
A-CTO [13]	Average	9.95E+06	139 917	3.50E+05	2.07E+06	3.10E+05	1.03E+05	6.55E+05	1.31E+05	3.74E+03	2.92E+03
	Deviation	0.53473	0.415846	0.952681	0.455927	0.161371	0.140878	0.843765	0.795779	0.109978	0.182851
	p-Value	0.039334	0.022559	0.047848	0.037958	0.03726	0.046979	0.049419	0.010017	0.040984	0.049886
	Time (s)	4456.65	4671.63	4713.01	4656.34	4758.8	4674.98	4738.1	10 539.8	10 671.7	10 844.8
KKO [16]	Average	100	1100.01	8105.72	1901.1	1700	1749.5	2249.5	2399.25	2599.25	2699.29
	Deviation	0	0	0.236849	0	0	0.573791	0.668844	0.603552	0.314019	0.040386
	p-Value	0.038592	0.044479	0.029881	0.001612	0.018366	0.035847	0.037613	0.022342	0.02878	0.003275
	Time (s)	946.708	1172.24	890.743	1192.6	1038.78	923.627	939.754	2200.34	2210.1	2262.09
BDO	Average	100	1100	902.3783	1900.77	1700	1749.18	2107.72	2300	2500	2601.22
	Deviation	0	0	10.19399	0.091651	0	0.011547	0.883346	0	0	37.91175
	Time (s)	857.022	957.138	1022.53	858.111	1236.84	1069.57	1251.12	2754.38	2771.8	2817.88

(move towards the driver dolphin) using (6) or search for a better solution within the search space (encircle other locations) using (7). This process of hunting makes it unique and helps to enhance their hunting. In case of the proposed BDO scheme, this process increases the diversity of search as at given instant of time, more number of solutions in the search space are explored. Hence, the searching ability and convergence speed of the algorithm is increased. This also decreases the chances of the algorithm to stuck into a local minima and increase the probability to converge to a global optimal solution.

- **Hunting strategy:** Once the location of the prey is decided, the driver dolphin starts to shrink the encirclement. During this time other dolphins of the group take random positions and move towards the prey location. This strategy helps to increase the harvest of the attack. This strategy of hunting is replicated using (8) and (9). This strategy in BDO helps to improve the exploitation capabilities. To further increase this capabilities, each variable which represents the dolphin position is randomly improved using (9) so that all close solutions can be examined. In this process, if any solution is found to be more reliable, it is termed as a global optimal solution.
- **Strategy randomizer operator:** Dolphins to increase its search diversity and harvest by switching between the two strategies, i.e., searching and hunting. During the searching strategy, some dolphins search random locations for the prey whereas some follow the driver dolphin as previously discussed. Whereas, during hunting, they shrink the encirclement in order to increase the harvest. This scheme are



(a) F_1 (b) F_4 (c) F_7 (d) F_9

Fig. 9. Convergence behavior of objective function considering 500*D* problem.

also replicated in the proposed BDO scheme. As discussed before, searching strategy replicates the exploration phases whereas hunting replicates the exploitation phase. To replicate the switching process in the proposed BDO scheme, a strategy randomizer operator is defined. The operator can be tuned in order to provide efficient time to explore and exploit the search space. This helps in maintaining a balance between exploration and exploitation phases.

- θ **operator:** To confine the search space, operator θ is used. This operator starts with a larger value and is gradually decreased to confine the search space. This helps to replicate the effect of encircling and shrinking of the search space as shown by Figs. 1 and 3. With the larger values initially, it helps in exploration of the search space. Whereas, with smaller value, it helps in the tuning of the obtained solution. This operator is similar to the convergence operator used in GWO, or the weight factor used in PSO.

These strategies together helps in enhancing the overall performance of the proposed BDO technique which also makes it superior to some of the other existing optimization schemes. This can be observed from the CASE STUDY A which presents a comparative study on the benchmark functions which are generally used to evaluate the exploration, exploitation and local minima avoidance performance of optimization techniques. The results obtained for these functions using the proposed BDO scheme is compared to some of the other existing optimization methods. From the study, it is observed that the proposed BDO scheme not only provides reliable and effective results for these functions, i.e. better average result but the overall stability of result is better, i.e. better standard deviation. Further, it is also observed that the speed at which the objective for these function converges is better when solved using the proposed BDO method. The computational burden to solve these function is also found to be better than some of the other existing optimization scheme. This proves that the proposed BDO technique is more effective than some of the existing methods used for comparison.

6. Case study B: Real-time problem

6.1. Introduction

In recent years, dynamic emission economic dispatch (DEED) problem has emerged as a multi-objective, non-convex problem related to power system. This problem aims to optimize the generating units of a conventional thermal power plant in such a way that the generation cost and emission rates are minimum. At the same time, power and physical constraints along with dynamically changing power demand should also be satisfied. Introduction of microgrid has further increased the complexity of this problem. Microgrids are small scale decentralized power systems which are combination of either conventional generators like thermal and diesel generator or renewable energy sources (RES) like solar power, wind power, etc. or both. RES have environmental and cost benefits over the conventional generators and are therefore, widely used in microgrid. Microgrid can operate either in islanded mode i.e individual operation or in connection to the grid. As the power output of the renewable sources are dynamically changing, the conventional generators are optimized to work accordingly.

Many methods have been proposed to solve this problem in past few years. In [32], authors presented a study to solve DEED problem with interconnected microgrid with techniques like PSO, DE, SOS, GWO, and WOA. The system considered for the study included a three generating unit, solar plant and wind plant. In [33], a modified harmony search algorithm is presented to solve this problem. The system considered also includes a three generating unit, solar plant and wind plant. In [34], authors presented a study for only DEED problem considering a five, six, ten, sixteen and thirty generating unit system. The authors used harmony search variant with constraint handling method to solve this problem. Some of the other literature available on this study are reduced gradient method [35], ant colony optimization [35], cuckoo search algorithm [36], interior search algorithm [36], evolution programming [37], pattern search (PS) optimization [38], time varying phasor particle swarm optimization [39], etc .

In this study, BDO technique is used to solve this DEED problem with interconnected microgrid. To show the effectiveness of

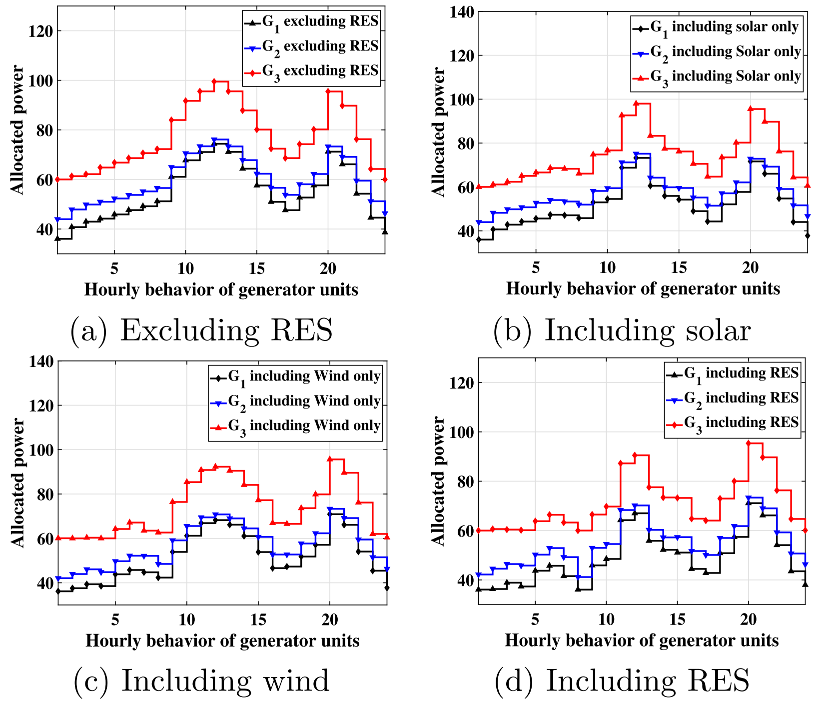


Fig. 10. Behavior of generating units considering DELD objective for TEST CASE I.

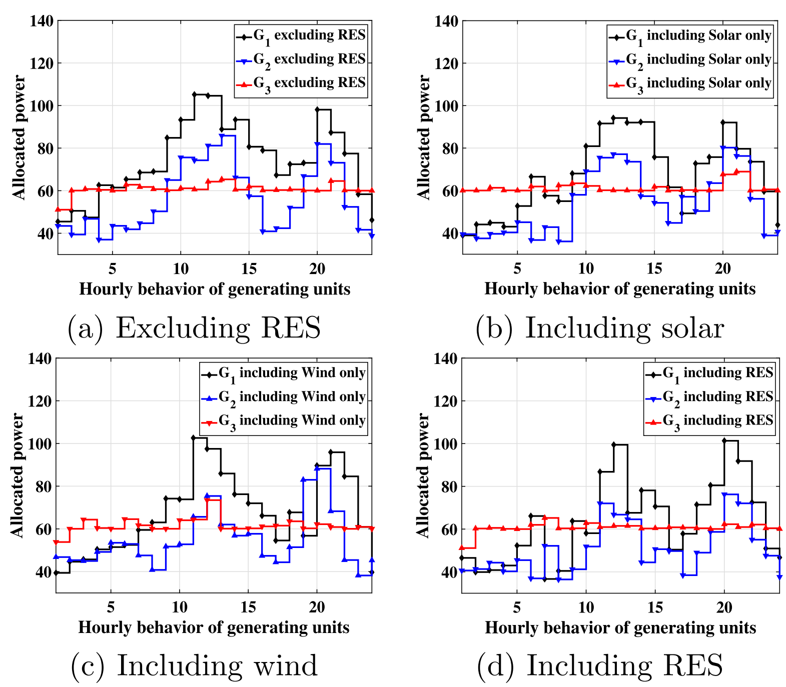


Fig. 11. Behavior of generating units considering DEED objective for TEST CASE I.

the proposed method, two test systems are considered in this study. The first system consists of three conventional generating units, whereas, the second system includes five generating units. The system also include two renewable sources which are solar farm with a capacity of 40 MW and wind farm with a capacity of 30 MW.

The next section presents a discussion on the problem formulation and constraints considered for DEED problem.

6.2. Problem formulation

6.2.1. Objective functions

Dynamic Economic Load Dispatch (DELD) problem:. In a DELD problem, all the conventional generating units are optimized in such a way that the overall generation cost is minimized. Apart from the minimization of the fuel cost, all the operating constraints and load demand should also be satisfied. The fuel cost for an *n* generator unit system operating for a time period of

Table 6
Parameter setting for different optimizers for TEST CASE I.

Method	Parameter setting
PSO[a] [32]	$\omega_{max} = 0.9$; $\omega_{min} = 0.4$; $c_1 = c_2 = 2$;
DE[a] [32]	$F = 0.7$; cross-over probability = 0.2
SOS[a] [32]	Benefit factor = 2
WOA[a] [32]	Convergence constant a = [2, 0]
GWO[a] [32]	Convergence constant a = [2, 0]
PPSO [40]	Phasor angle θ = [0, 2π]
KKO [16]	Endurance factor e = [2, 0]
BDO	$a_f = 3$, $\theta_{min} = 0.2$; $S_r = 0.8$

[a]Parameters are reported in [32].

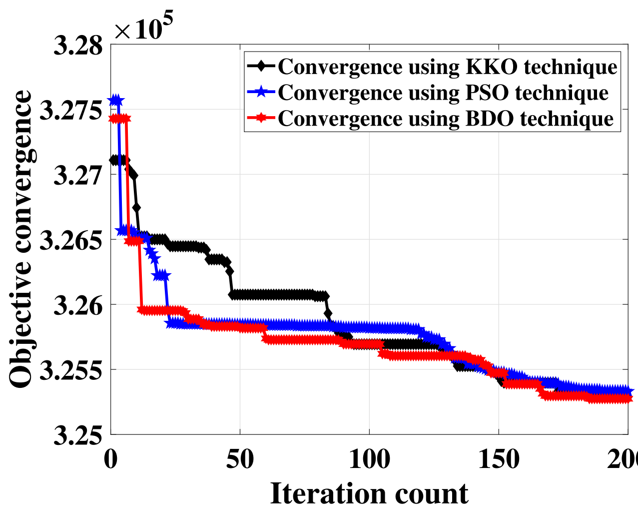


Fig. 12. Convergence behavior of objective function for TEST CASE-I.

24 hours can be computed using a quadratic equation represented as:

$$G_{cost} = \sum_{t=1}^{24}\sum_{i=1}^{n}(a_iP_i^2 + b_iP_i + c_i), \tag{16}$$

where G_{cost} is total generation cost, t represents the time instant in hours, P_i is the active power generated by ith generating unit, a_i, b_i and c_i are the cost coefficients associated with the ith generating unit.

But, with the introduction of valve point effect in a practical case scenario, this quadratic equation is replaced with a nonlinear equation which is represented as follows.

$$G_{cost} = \sum_{t=1}^{24}\sum_{i=1}^{n}(a_iP_i^2 + b_iP_i + c_i + d_i(e_i(sin(P_{min} - P_i)))), \tag{17}$$

where G_{cost} is the overall generation cost, P_{min} is the minimum power capacity of ith generating unit, d_i and e_i are the coefficients associated with the ith generating unit due to valve point effect.

Dynamic Economic Emission Dispatch (DEED) problem:. Considering the environmental problem caused by the burning of fossil fuel in a thermal power plant, power industries try to curb the emission rate. Emission of harmful gases from n generator unit system operating for a time period of 24 hours can be computed as following:

$$E = \sum_{t=1}^{24}\sum_{i=1}^{n}(u_iP_i^2 + \upsilon_iP_i + w_i + x_ie^{(y_iP_i)}), \tag{18}$$

where E is the total emission, u_i, υ_i, w_i, x_i and y_i are the emission coefficients associated with the ith generating unit.

Combining the two objectives i.e. minimization of fuel cost and emission rate is deal with the problem of DEED. Considering the two objectives, DEED problem can be referred as a multi-objective problem which is mathematically modeled as follows.

$$F = G_{cost} + h_l \times E, \tag{19}$$

where F is the overall objective which has to be minimized, h_l is referred as the penalty factor and is computed as presented in [32].

Renewable Energy Source (RES) integration:. RES are the clean energy sources which neither include fuel cost nor emits harmful gases. The cost associated with RES are installation and maintenance cost which can be mathematically represented as follows.

$$C_{RES} = P_{RES}\left(\frac{r}{1-(1+r)^{-N}} \times I_p + G_e\right), \tag{20}$$

where C_{RES} is the total cost, r is interest scale, N is investment duration in years, I_p is ratio of investment cost to established power and G_e is the operation and maintenance cost.

For this study, the islanded microgrid includes a wind farm and solar farm. The data for r, N, I_p and G_e are taken from [45]. Using the data, the cost for a solar farm is given as $F_{pv} = 547.7483 \times P_{pv}$ and same for a wind farm is given as $F_{wind} = 153.3810 \times P_{wind}$. Finally, consider all objective together, the objective for a DEED problem with an integrated renewable microgrid is given as follows.

$$\begin{aligned} F_{cost} = &\sum_{t=1}^{24}\sum_{i=1}^{n}(a_iP_i^2 + b_iP_i + c_i + d_ie_i(sin(P_{min} - P_i)) \\ &+ h_l \times (u_iP_i + \upsilon_iP_i + w_i + x_ie^{(y_iP_i)}) \\ &+ 547.7483 \times P_{pv} + 153.3810 \times P_{wind}), \end{aligned} \tag{21}$$

6.2.2. Operating constraints

Generator constraint:. All the conventional generating units as well as RES must be operated within the prescribed limits of operation. This limit is mathematically represented as follows.

$$P_i^{min} \le P_i \le P_i^{max}, \tag{22}$$

$$P_{RES}^{min} \le P_{RES} \le P_{RES}^{max}, \tag{23}$$

where P_i^{min} and P_i^{max} is the minimum and maximum power limits of ith generating unit, P_{RES}^{min} and P_{RES}^{max} is the minimum and maximum power limits for RES.

Power balance constraint:. According to the power balance criterion, at any time instant the power generated by the conventional generating units and RES should satisfy the overall power demand. Mathematically, power balance criterion is represented as:

$$P_{load} = P_G + P_{RES}, \tag{24}$$

where P_{load} is the load demand, P_G is the generated power by the conventional generating units and P_{RES} is the generated power from the RES units.

Under practical operation scenario, losses are always associated with the transmission of power. Considering transmission losses, power balance criterion is represented as follows:

$$P_{load} + P_{loss} = P_G + P_{RES}, \tag{25}$$

where P_{loss} is the power loss in transmission. It is computed as

$$P_{loss} = \sum_{i=1}^{n}\sum_{j=1}^{n}P_iB_{ij}P_j + \sum_{i=1}^{n}B_iP_i + B_0, \tag{26}$$

where B_{ij}, B_i and B_0 are the loss coefficients.

Table 7
Comparative analysis of the cost function obtained for TEST CASE-I.

Technique	ELD Problem				CEED Problem			
	Without RES	With wind only	With solar only	With RES	Without RES	With wind only	With solar only	With RES
PSO [32]	176 177.9175	272 045.2086	204 025.1856	299 919.4357	202 886.6496	297 912.8001	230 029.0775	325 377.3173
DE [32]	176 169.0719	272 036.3530	204 006.9307	299 916.0487	202 884.8852	297 911.5005	230 024.3813	325 371.3072
SOS [32]	176 168.04244	272 034.5209	204 001.6485	299 906.3846	202 882.0837	297 910.2332	230 023.7559	325 369.7976
GWO [32]	176 167.8827	272 033.5531	203 988.3084	299 896.6562	202 882.6042	297 908.2971	230 020.3064	325 368.4448
WOA [32]	176 166.5662	272 031.0549	203 987.5104	299 895.7531	202 881.7751	297 907.5634	230 019.0483	325 364.4919
PPSO [40]	176 168	272 032	203 988	299 897	202 890	297 906	230 018	325 359
KKO [16]	176 167.561	272 030.9569	203 986.9412	299 897.0159	202 876.5959	297 892.159	230 018.48	325 355.69
BDO	176 166	272 029.7	203 985.8	299 895.1	202 874.7	297 890.5	230 014.9	325 351.6

Table 8
Computational burden to solve TEST CASE I using different optimization techniques (s).

Methods	DELD				DEED			
	Excluding RES	Including wind	Including solar	Including RES	Excluding RES	Including wind	Including solar	Including RES
PSO [32]	–	–	–	–	–	–	–	–
DE [32]	–	–	–	–	–	–	–	–
SOS [32]	–	–	–	–	–	–	–	–
GWO [32]	–	–	–	–	–	–	–	–
WOA [32]	–	–	–	–	–	–	–	–
PPSO [40]	137.1179	138.3388	139.4098	141.3679	145.9679	147.3661	149.2688	152.689
KKO [40]	131.9517	133.9452	135.3338	138.43	141.9097	144.2302	145.7859	147.8944
BDO [40]	134.3685	135.6806	137.3217	140.6445	144.7075	145.4985	147.5748	150.0002

Table 9
Parameter setting for different optimizers for TEST CASE I.

Method	Parameter setting
DE-SQP[a] [41]	Population = 60; F = 0.423; CR = 0.885
PSO-SQP[a] [41]	Population = 60; ω_{max} = 0.9; ω_{min} = 0.4; $c_1 = c_2$ = 2.25
MODE [42]	–
MOHDE-SAT [42]	–
NPAHS [43]	–
EP [37]	–
SA [38]	–
PS [38]	–
MHS[b] [34]	Harmony Memory rate (HMR) - 0.9; Pitch Adjusting Rate (PAR)-0.4; Bandwidth (BW) = 0.0001
HIS[b] [34]	HMR = 0.95; PAR_{max} = 0.99; PAR_{min} = 0.35; BW_{max} = 0.1; BW_{min} = 0.0001
DHS[b] [34]	HMR_{max} = 0.95; HMR_{min} = 0.8; PAR_{max} = 0.5; PAR_{min} = 0.2; BW_{max} = 0.55; BW_{max} = 0.15
HS-NPSA[b] [34]	HMR = [0.9, 1]; PAR = [0,0.5]; BW = [0,1]
NEHS[b] [34]	HMR = 0.1-1/N; PAR = 0.5; BW = 0.05
PSO [30]	ω_{max} = 0.9; ω_{min} = 0.4; $c_1 = c_2 = 2$
GWO [8]	Convergence constant a = [2, 0]
FPA [44]	Probability switching p=0.8
PPSO [40]	Phasor angle θ = [0, 2π]
KKO [16]	Endurance factor e = [2, 0]
BDO	a_f = 3, θ_{min} = 0.2; S_r = 0.8

[a]Parameter reported in [41];
[b]Parameter reported in [34].

6.3. Result and discussions

The proposed BDO technique is used to solve a DEED with integrated microgrid problem related to power systems. To evaluate the performance of BDO technique, two test case are considered.

6.3.1. TEST case I: 3 generating unit system

For the first test case, a three generating unit system is considered. The dynamic demand, solar, wind power outputs and generator fuel and emission coefficients are taken form [32]. For this system, transmission losses are neglected. The system is tested for two objectives (i) considering DELD problem (ii) considering DEED problem. During this tests four combination which are (i) without RES, (ii) with RES, (iii) with solar units and (iv) with wind units are evaluated. All test cases are simulated for ten times and the best results are further analyzed. For a comparative analysis and to show the supremacy of the proposed BDO scheme, some existing optimization schemes such as PSO [32], GWO [32], WAO [32], symbiotic organism search (SOS) [32], DE [32], phasor PSO (PPSO) [40], and KKO [16] are considered in this study. The result for PSO, GWO, WAO, SOS, and DE are taken from [32]. Whereas, the optimal results for PPSO, and KKO are obtained considering the same system settings i.e. population size 30 and iteration count 1000 as set in [32]. Further, the parameter setting for these methods along with BDO scheme is presented in Table 6.

Considering a DELD problem objective, the hourly behavior of the conventional generating units for all the test scenarios are presented in Fig. 10. The overall generation cost obtained using BDO is compared with some existing results obtained using PSO, GWO, WAO, SOS, DE, PPSO, and KKO. For a comparative analysis, these results are presented in Table 7. It is observed that the proposed BDO technique provides a better optimized result respect to WOA, GWO, SOS, DE, PSO, PPSO, and KKO. The computational burden to solve the test case is further reported in Table 8. The average computational time revels that the proposed BDO scheme takes less time to solve the test case in comparison to some of the other optimization schemes. Further, considering a DEED problem objective, the hourly behavior of the conventional generating units for all the test scenarios are presented in Fig. 11. The overall generation cost obtained using BDO is compared with some existing results. For a comparative analysis, these results are presented in Table 7. It is observed that the proposed BDO technique provides a better optimized result with respect to WOA, GWO, SOS, DE, PSO, PPSO, and KKO. Proposed BDO scheme provides the optimal result in approximate 140 seconds for all the test scenario as discussed in Table 8 which is better than some of the other optimizers used for comparison. The convergence behavior of the fuel cost obtained using BDO, PPSO, and KKO for this problem considering all sources together is presented in Fig. 12. From the convergence plot, it is clear that BDO shows faster convergence characteristics in comparison to PPSO, and KKO.

Table 10
Comparative analysis of the cost function obtained for TEST CASE-II.

Problem	DELD problem				DEED problem							
Cases	Without RES	With wind only	With solar only	With RES	Without RES		With wind only		With solar only		With RES	
Techniques					Emission	Fuel cost	Emission	Fuel cost	Emission	Fuel cost	Emission	Fuel cost
DE-SQP [41]	43 161	–	–	–	44 450	19 616	–	–	–	–	–	–
PSO-SQP [41]	43 263	–	–	–	44 542	19 772	–	–	–	–	–	–
MODE [42]	46 747	–	–	–	47 330	18 116	–	–	–	–	–	–
MOHDE-SAT [42]	46 478	–	–	–	48 214	18 011	–	–	–	–	–	–
NPAHS [43]	43 072.99	–	–	–	45 196	18 630	–	–	–	–	–	–
EP [37]	46 777	–	–	–	48 628	21 154	–	–	–	–	–	–
SA [38]	47 356	–	–	–	48 621	21 188	–	–	–	–	–	–
PS [38]	46 530	–	–	–	47 911	18 927	–	–	–	–	–	–
HIS [34]	44 097.05446	–	–	–	46 366.62	18 330.76921	–	–	–	–	–	–
DHS [34]	45 890.84373	–	–	–	47 960.86	18 377.47801	–	–	–	–	–	–
HS-NPSA [34]	43 927.30521	–	–	–	46 466.36	18 293.25003	–	–	–	–	–	–
MHS [34]	45 497.74	–	–	–	47 390.96	18 423.77597	–	–	–	–	–	–
NEHS [34]	43 066.07313	–	–	–	45 398.01	18 392.33716	–	–	–	–	–	–
PSO [46]	47 852	150 479	83 203.2	177 868	50 893	20 163	51 572.1	20 384.5	51 137.9	20 720.7	50 198.5	19 606.4
FPA [44]	47 864.5	147 618	81 102.2	177 924	51 517.6	20 614.3	49 927.3	20 517.4	50 920.2	20 719.7	49 898.7	19 714.6
GWO [8]	45 194.1	145 991	77 705.7	177 892	47 356.1	20 234.9	47 328	19 042	48 325.3	18 325.3	48 254.3	18 471.6
PPSO [40]	46 363.9	147 033	78 494.8	177 858	47 832.9	20 030.5	46 932.8	19 737.2	47 185.6	20 076.9	46 649.4	18 686.5
KKO [16]	44 227.5	145 981	77 201.5	177 834	46 825.4	20 071.3	46 911.8	19 285.4	46 527.5	19 745	46 651.3	18 452.2
BDO	42 828.6	144 739	77 156.8	177 814	44 538.3	20 910.2	46 742.9	19 884.6	44 434.9	20 977.8	45 505.8	19 160.7

Table 11
Computational burden to solve TEST CASE II using different optimization techniques (s).

Methods	DELD				DEED			
	Excluding RES	Incuding wind	Wincluing solar	Including RES	Excluding RES	Incuding wind	Wincluing solar	Including RES
DE-SQP [41]	–	–	–	–	–	–	–	–
PSO-SQP [41]	–	–	–	–	–	–	–	–
MODE [42]	–	–	–	–	–	–	–	–
MOHDE-SAT [42]	–	–	–	–	–	–	–	–
NPAHS [43]	–	–	–	–	–	–	–	–
EP [37]	–	–	–	–	–	–	–	–
SA [38]	–	–	–	–	351.98	–	–	–
PS [38]	514.25	–	–	–	272.2	–	–	–
HIS [34]	46.18	–	–	–	49.18	–	–	–
DHS [34]	52.76	–	–	–	56.73	–	–	–
HS-NPSA [34]	45.71	–	–	–	49.08	–	–	–
MHS [34]	51.80	–	–	–	62.26	–	–	–
NEHS [34]	41.64	–	–	–	47.36	–	–	–
PSO [30]	280.2809	281.2445	281.9924	282.5689	289.403	290.4814	292.4605	294.9916
GWO [8]	262.9463	263.5226	264.4008	266.3411	271.6928	272.2768	274.6313	277.8842
FPA [44]	267.2972	268.0375	269.6288	270.0623	275.6027	276.4263	277.6272	278.0355
PPSO [40]	280.071	280.9388	281.414	282.5286	287.0388	287.7824	288.4922	289.1147
KKO [16]	245.464	245.614	246.1346	246.6294	251.7809	251.6717	252.1152	254.4435
BDO	250.3877	251.939	252.9954	252.3034	257.1256	257.5678	258.5657	259.8972

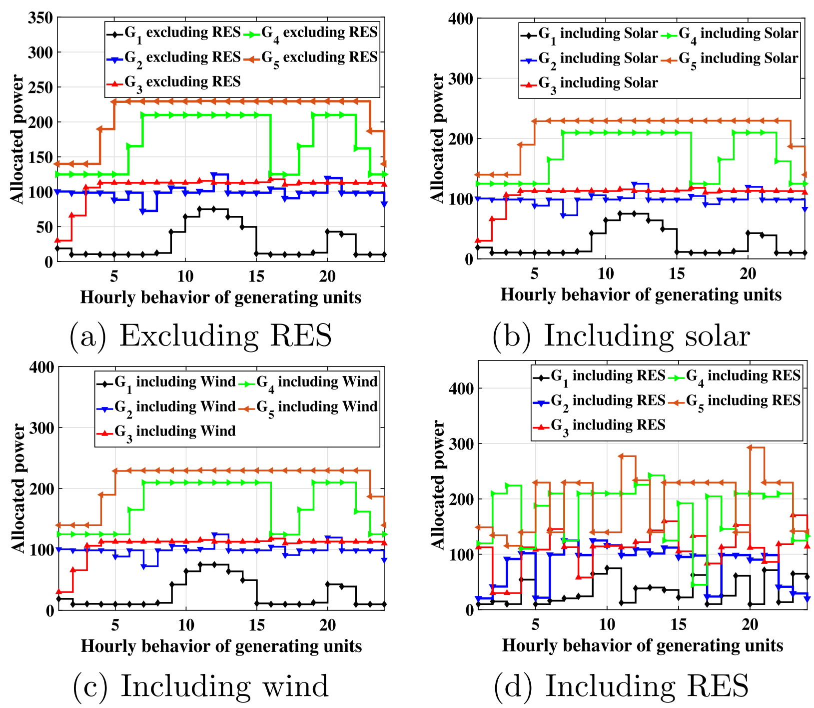
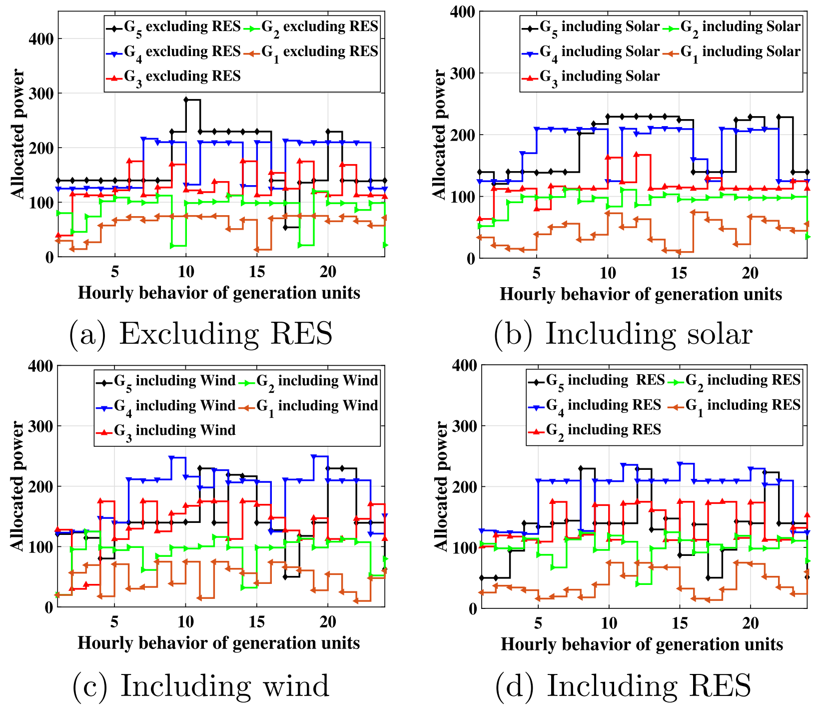


Fig. 13. Behavior of generating units considering DELD objective for TEST CASE II.

6.3.2. TEST case II: 5 generating units system

For this test case, a five conventional generating system is considered. The cost and emission coefficient for the generating units and dynamic load demand are taken from [46]. The islanded microgrid characteristics is considered to be same as that used for TEST CASE I. This system is also simulated with same objectives i.e. considering DELD problem and considering DEED problem with four conditions i.e. without RES, with RES, with wind and with solar. To show the supremacy of the proposed BDO scheme to solve this test system, a comparative study using some existing optimization schemes such as hybrid DE and sequential quadratic programming (DE-SQP) [41], hybrid PSO and DWP (PSO-SQP) [41], multiobjective differential evolution algorithm (MODE) [42], multi-objective hybrid differential evolution with simulated annealing technique (MOHDE-SAT) [42], efficient harmony search approach based on a new pitch adjustment rule (NPAHS) [43], EP [37], simulated annealing (SA) [38], PS [38], improved harmony search (IHS) [34], differential harmony search (DHS) [34], harmony search with a novel parameter setting approach (HS-NPSA) [34], modified harmony search algorithm (MHS) [34], new enhanced harmony search (NEHS) [34], PSO [30], GWO [8], PPSO [40], and KKO [16] are considered in this study. The result for DE-SQP, PSO-SQP, MODE, MOHDE-SAT, NPAHS, EP, SA, PS, HIS, DHS, HS-NPSA, MHS, and NEHS are taken from [34], whereas the in case of PPSO, PSO, GWO, FPA and KKO the optimal results are obtained by considering the system settings i.e. population size 30 and iteration count 1000. Further, the other parameter setting for these methods along with BDO scheme used to obtain the optimal result is presented in Table 9. With the set parameters, all the different test cases are simulated for ten times and the best results are further analyzed for the superiority of the proposed technique.



(a) Excluding RES (b) Including solar

(c) Including wind (d) Including RES

Fig. 14. Behavior of generating units considering DEED objective for TEST CASE II.

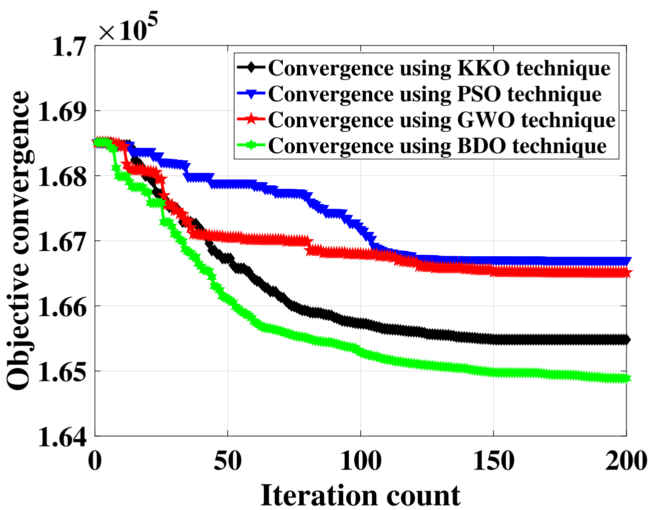


Fig. 15. Convergence behavior of objective function for different optimizer for TEST CASE II.

For this test case, considering the DELD objective, the best hourly behavior of the conventional generating units for all the test cases are presented in Fig. 13. For a comparative study, the overall generation cost obtained using BDO is compared with some existing results obtained using other existing techniques. These results are presented in Table 10. It is observed that the proposed BDO technique provides a better optimized result. Moreover, based on the Table 10, the computational time to solve the objective is found to be less as compared to some of the other existing optimization schemes taken into consideration. Similarly, the best hourly behavior of the conventional generating units for all the test cases, considering DEED problem, is presented in Fig. 14. Result obtained is compared with result of some other existing optimization techniques to show the superiority of proposed BDO technique. This result is presented in Table 10. It is observed that the proposed BDO technique provides a better result by optimizing the fuel cost more efficiently. Further, the computation burden to solve this problem is reported in Table 11. From the results it is evident that the proposed BDO scheme takes less time to solve this test case in respect to some of the other existing optimization schemes in considerations. To study the convergence behavior of the cost function obtained using the proposed technique, PSO, GWO, PPSO, and FPA for this problem considering all sources is presented in Fig. 15. It is clearly observed that, BDO provides a faster convergence with respect to other optimizers used for comparison.

Hence, it can be concluded after observing all the test cases results that the proposed BDO technique is sufficient in solving both large and small system related to power system problems. Moreover, the proposed BDO method can be used for solving real-time optimization problems.

7. Conclusion

In this article, a bottlenose dolphin optimizer (BDO) is proposed. The proposed BDO is a population-based meta-heuristic optimization technique which mimics the mud ring hunting behavior of the bottlenose dolphins. To show the superiority of the proposed technique over other existing techniques, two different types of test cases are considered. The first test case comprises of benchmark functions to examine the exploration, exploitation, convergence and local minima avoidance capabilities. It is observed from the results that the proposed optimization technique is competitive enough with respect to other existing techniques. The next test case, i.e. dynamic emission economic dispatch problem is used to test the effectiveness of proposed technique to solve real-time optimization problems. The results proves that BDO can effectively solve this problem by optimizing the generating units to provide a better optimized fuel cost and emission rates.

Further, it can be concluded that proposed BDO technique can be used to solve real-time optimization problems related to the field of engineering, economics, science, etc.

CRediT authorship contribution statement

Abhishek Srivastava: Conceptualization, Methodology, Writing – original draft. **Dushmanta Kumar Das:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J.H. Holland, Genetic algorithms, Sci. Am. 267 (1) (1992) 66–73.
[2] H.-G. Beyer, H.-P. Schwefel, Evolution strategies–a comprehensive introduction, Nat. Comput. 1 (1) (2002) 3–52.
[3] D. Dasgupta, Z. Michalewicz, Evolutionary Algorithms in Engineering Applications, Springer Science & Business Media, 2013.
[4] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, Gsa: a gravitational search algorithm, Inform. Sci. 179 (13) (2009) 2232–2248.
[5] A. Hatamlou, Black hole: A new heuristic optimization approach for data clustering, Inform. Sci. 222 (2013) 175–184.
[6] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Micro Machine and Human Science, 1995. MHS'95. Proceedings of the Sixth International Symposium on, IEEE, 1995, pp. 39–43.
[7] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.
[8] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[9] S. Shadravan, H. Naji, V.K. Bardsiri, The sailfish optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems, Eng. Appl. Artif. Intell. 80 (2019) 20–34.
[10] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, Future Gener. Comput. Syst. 97 (2019) 849–872.
[11] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: A bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191.
[12] P. Das, D.K. Das, S. Dey, A new class topper optimization algorithm with an application to data clustering, IEEE Trans. Emerg. Top. Comput. (2018).
[13] A. Srivastava, D.K. Das, A new aggrandized class topper optimization algorithm to solve economic load dispatch problem in a power system, IEEE Trans. Cybern. (2020) 1–11, http://dx.doi.org/10.1109/TCYB.2020.3024607.
[14] R.V. Rao, V.J. Savsani, D. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (3) (2011) 303–315.
[15] A.H. Kashan, League championship algorithm: a new algorithm for numerical function optimization, in: Soft Computing and Pattern Recognition, 2009. SOCPAR'09. International Conference of, IEEE, 2009, pp. 43–48.
[16] A. Srivastava, D.K. Das, A new kho-kho optimization algorithm: An application to solve combined emission economic dispatch and combined heat and power economic dispatch problem, Eng. Appl. Artif. Intell. 94 (2020) 103763.
[17] F. Glover, Tabu search and adaptive memory programming—advances, applications and challenges, in: Interfaces in Computer Science and Operations Research, Springer, 1997, pp. 1–75.
[18] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82.
[19] K. Sörensen, Metaheuristics—the metaphor exposed, Int. Trans. Oper. Res. 22 (1) (2015) 3–18.
[20] T.-q. Wu, M. Yao, J.-h. Yang, Dolphin swarm algorithm, Front. Inf. Technol. Electron. Eng. 17 (8) (2016) 717–729.
[21] L. Marino, Cetacean brain evolution: multiplication generates complexity, Int. J. Comp. Psychol. 17 (1) (2004).
[22] M. Krützen, S. Kreicker, C.D. MacLeod, J. Learmonth, A.M. Kopps, P. Walsham, S.J. Allen, Cultural transmission of tool use by indo-pacific bottlenose dolphins (tursiops sp.) provides access to a novel foraging niche, Proc. R. Soc B: Biol. Sci. 281 (1784) (2014) 20140374.
[23] J. Mann, R.C. Connor, P.L. Tyack, H. Whitehead, Cetacean Societies: Field Studies of Dolphins and Whales, University of Chicago Press, 2000.
[24] B.D. López, J.A.B. Shirai, Marine aquaculture and bottlenose dolphins'(tursiops truncatus) social structure, Behav. Ecol. Sociobiol. 62 (6) (2008) 887.
[25] H. Shirihai, B. Jarrett, G.M. Kirwan, Whales, Dolphins, and Other Marine Mammals of the World, Princeton University Press, 2006.
[26] J. Weiss, Foraging habitats and associated preferential foraging specializations of bottlenose dolphin (tursiops truncatus) mother-calf pairs, Aquat. Mamm. 32 (1) (2006) 10.
[27] E.E. Duffy-Echevarria, R.C. Connor, D.J. St. Aubin, Observations of strand-feeding behavior by bottlenose dolphins (tursiops truncatus) in bull creek, south carolina, Mar. Mamm. Sci. 24 (1) (2008) 202–206.
[28] A. Berta, Whales, Dolphins, and Porpoises: A Natural History and Species Guide, University of Chicago Press, 2015.
[29] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359.
[30] J. Kennedy, R. Eberhart, Particle swarm optimization (pso), in: Proc. IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942–1948.
[31] R. Eberhart, J. Kennedy, Particle swarm optimization, in: Proceedings of the IEEE International Conference on Neural Networks, Vol. 4, Citeseer, 1995, pp. 1942–1948.
[32] B. Dey, S.K. Roy, B. Bhattacharyya, Solving multi-objective economic emission dispatch of a renewable integrated microgrid using latest bio-inspired algorithms, Eng. Sci. Technol. Int. J. 22 (1) (2019) 55–66.
[33] E.E. Elattar, Modified harmony search algorithm for combined economic emission dispatch of microgrid incorporating renewable sources, Energy 159 (2018) 496–507.
[34] Z. Li, D. Zou, Z. Kong, A harmony search variant and a useful constraint handling method for the dynamic economic emission dispatch problems considering transmission loss, Eng. Appl. Artif. Intell. 84 (2019) 18–40.
[35] I.N. Trivedi, D.K. Thesiya, A. Esmat, P. Jangir, A multiple environment dispatch problem solution using ant colony optimization for micro-grids, in: 2015 International Conference on Power and Advanced Control Engineering (ICPACE), IEEE, 2015, pp. 109–115.
[36] I.N. Trivedi, P. Jangir, M. Bhoye, N. Jangir, An economic load dispatch and multiple environmental dispatch problem solution with microgrids using interior search algorithm, Neural Comput. Appl. 30 (7) (2018) 2173–2189.
[37] M. Basu, Dynamic economic emission dispatch using evolutionary programming and fuzzy satisfying method, Int. J. Emerg. Electr. Power Syst. 8 (4) (2007).
[38] J. Alsumait, M. Qasem, J. Sykulski, A. Al-Othman, An improved pattern search based algorithm to solve the dynamic economic dispatch problem with valve-point effect, Energy Convers. Manage. 51 (10) (2010) 2062–2067.
[39] S. Khatsu, A. Srivastava, D.K. Das, Solving combined economic emission dispatch for microgrid using time varying phasor particle swarm optimization, in: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 411–415, http://dx.doi.org/10.1109/ICACCS48705.2020.9074321.
[40] M. Gholamghasemi, E. Akbari, M.B. Asadpoor, M. Ghasemi, A new solution to the non-convex economic load dispatch problems using phasor particle swarm optimization, Appl. Soft Comput. 79 (2019) 111–124.
[41] A. Elaiw, X. Xia, A. Shehata, Hybrid de-sqp and hybrid pso-sqp methods for solving dynamic economic emission dispatch problem with valve-point effects, Electr. Power Syst. Res. 103 (2013) 192–200.
[42] H. Zhang, D. Yue, X. Xie, S. Hu, S. Weng, Multi-elite guide hybrid differential evolution with simulated annealing technique for dynamic economic emission dispatch, Appl. Soft Comput. 34 (2015) 312–323.
[43] Q. Niu, H. Zhang, K. Li, G.W. Irwin, An efficient harmony search with new pitch adjustment for dynamic economic dispatch, Energy 65 (2014) 25–43.
[44] X.-S. Yang, Flower pollination algorithm for global optimization, in: International Conference on Unconventional Computing and Natural Computation, Springer, 2012, pp. 240–249.
[45] N. Augustine, S. Suresh, P. Moghe, K. Sheikh, Economic dispatch for a microgrid considering renewable energy cost functions, in: 2012 IEEE PES Innovative Smart Grid Technologies (ISGT), IEEE, 2012, pp. 1–7.
[46] M. Basu, Particle swarm optimization based goal-attainment method for dynamic economic emission dispatch, Electr. Power Compon. Syst. 34 (9) (2006) 1015–1025.