An Improved Marine Predators Algorithm for Optimal Reactive Power Dispatch With Load and Wind-Solar Power Uncertainties

JIAN ZHAO\textsuperscript{1}, LIYING LI\textsuperscript{1}, JIAxin YU\textsuperscript{2}, AND ZHIWU LI\textsuperscript{3}, (Fellow, IEEE)

\textsuperscript{1}School of Science, University of Science and Technology Liaoning, Liaoning, Anshan 114051, China
\textsuperscript{2}Hitachi Building Technology (Guangzhou) Company Ltd., Guangzhou 500650, China
\textsuperscript{3}Institute of Systems Engineering, Macau University of Science and Technology, Macau, Taipa 999078, China

Corresponding author: Zhiwu Li (zwli@must.edu.mo)

This work was supported in part by the National Natural Science Foundation of China under Grant U1731128; in part by the Natural Science Foundation of Liaoning Province, China, under Grant 2019-MS-174; in part by the Foundation of Liaoning Province Education Administration, China, under Grant 2019LNJIC12, Grant 2020LNQNO5, and Grant LJKZ0279; and in part by the Guangzhou Innovation and Entrepreneurship Leading Team Project under Grant 202009020008.

\section*{ABSTRACT}

With the growing usage of renewable energy resources, the fluctuation and randomness of solar and wind power output have a significant influence on the safe and stable operation of a power system. Deciding how to deal with these uncertainties has become a hot topic for researchers and practitioners. In this paper, a hybrid improved marine predators algorithm is proposed to solve the optimal reactive power dispatch problem with load demand, and wind-solar power uncertainties. The $\varepsilon$-constraint handling technique is adopted to deal with the constraints in the optimization problem. The uncertainty factors in the system are modeled using appropriate probability density functions. The backward reduction method is used to determine a representative set of scenarios from a large number of scenarios produced by a Monte Carlo simulation method. IEEE 30-bus test system is utilized to verify the proposed algorithm. The simulation results demonstrate that the algorithm can produce better results than other advanced metaheuristic algorithms.

\section*{INDEX TERMS}
The $\varepsilon$-constraint handling, marine predators algorithm, optimal reactive power dispatch, renewable energy, uncertainty.

\section*{I. INTRODUCTION}

The optimal reactive power dispatch (ORPD) problem is one of the significant research directions in the field of power system optimization. The primary objective of the ORPD problem is to maximize the voltage stability by optimizing the generator terminal voltages, transformer taps, shunt VAR compensators, and other equipment to minimize active power loss and voltage deviation [1], [2]. From the mathematical viewpoint, the control variables of the power system include both continuous variables (i.e., generator voltages) and discrete variables (i.e., tap of transformers and shunt VAR compensators), which make the ORPD problem difficult to solve. Therefore, seeking a better solution has received much attention.

In the early stages, many traditional mathematical methods, including the newton method [3] and linear programming [4], have been widely employed to address the ORPD problem. Unfortunately, these techniques have weaknesses that cannot be ignored, in particular poor convergence, complex computation, and inappropriate handling of discrete variables. To better solve the non-linear ORPD problem, scholars have developed numerous intelligent optimization techniques, like differential evolution (DE) [5], grey wolf optimizer (GWO) [6], cuckoo search algorithm (CS) [7], bat algorithm (BA) [8], and social spider optimization (SSO) [9].

Nowadays, with the increasing use of renewable energy resources (RERs) in the power system, the uncertainties
of wind power and photovoltaic power output has brought serious challenges to safe and stable operation. To realize efficient management, the uncertainty of the system must be taken into account. Therefore, the solutions of ORPD incorporating RERs are studied in [10], [11], [12], and [13]. The marine predators algorithm (MPA) is a recently proposed algorithm by Faramarzi et al. [14]. The research demonstrates that while the MPA performs favorably with different algorithms, it still has poor convergence and local optima traps when solving complex problems [15], [16], [17].

In this paper, considering the uncertainties of load demand and RERs, a hybrid improved marine predators algorithm (HIMPA) is applied to solve the ORPD problem. The HIMPA has been modified in the following three aspects: i) to improve the quality of the initial solution, an opposition-based learning method is adopted; ii) to balance the exploration and development capabilities, the mid-term stage of traditional MPA is replaced by the modified DE/best – to-rand/1 strategy; iii) SS operator is used to enhance the development stage of the algorithm and accelerate the convergence speed. Since the constraints are difficult to deal with, we use the $\varepsilon$-constraint handling [18] and HIMPA integration technology to optimize the decision variables in the system. Moreover, the uncertainty in the power system is also modeled using various probability density functions (PDFs) [11]. In addition, 1000 scenarios are generated employing the Monte Carlo Simulation method [19] and then chosen by applying a backward reduction technique. Finally, the feasibility and effectiveness of the HIMPA algorithm are verified on IEEE 30-bus test system.

The remainder of the paper is structured as follows: Section II describes the ORPD problem formulation and $\varepsilon$-constraint handling technique. Section III describes the uncertainties model of RERs and load demand. In Section IV, the background knowledge is briefly recalled. The improved MPA is detailed in Section V. The related experimental analysis and discussion are provided in Section VI. Section VII concludes the work.

II. PROBLEM FORMULATION
The mathematical model of the ORPD problem can generally be defined as follows:

$$\text{Min : } F(\mathbf{x}, \mathbf{u})$$

Subject to: $g_i(\mathbf{x}, \mathbf{u}) \leq 0, \quad i = 1, 2, \cdots, m$  \hspace{1cm} (1)

$h_j(\mathbf{x}, \mathbf{u}) = 0, \quad j = m+1, \cdots, p$  \hspace{1cm} (2)

where $F(\mathbf{x}, \mathbf{u})$ is the objective function, $g_i(\mathbf{x}, \mathbf{u})$ and $h_j(\mathbf{x}, \mathbf{u})$ indicate the $i$th inequality constraint and the $j$th equality constraint, respectively. The vectors $\mathbf{x}$ and $\mathbf{u}$ represent control and state variables, respectively, i.e.,

$$\mathbf{x} = [V_{G_1}, \cdots, V_{G_{NG}}, Q_{C_1}, \cdots, Q_{C_{NC}}, T_1, \cdots, T_{NT}]$$

$$\mathbf{u} = [V_{L_1}, \cdots, V_{L_{NL}}, Q_{G_1}, \cdots, Q_{G_{NG}}, S_{h_1}, \cdots, S_{l_{id}}]$$

where $V_{G_i}$ represents the voltage at the $i$th generator bus, $Q_C$ denotes the shunt compensation at the $j$th bus and $T_k$ signifies the $k$th branch transformer tap; $NG, NC,$ and $NT$ are the number of generators buses, shunt compensators, and transformers, respectively. In this paper, $Q_C$ and $T_k$ are discrete variables. $V_{L_m}$ is the voltage at the $m$th load bus, $Q_{Gi}$ denotes the reactive power at the $i$th generator bus and $S_{l_{ij}}$ represents the line loading at the $q$th line. $NL$ and $nl$ are the number of load buses and the transmission lines, respectively.

A. OBJECTIVE FUNCTIONS
1) MINIMIZATION OF THE REAL POWER LOSS
The mathematical model with the objective of reducing the real power loss ($F_{\text{Loss}}$) can be expressed as [20]:

$$F_{\text{Loss}} = \sum_{i=1}^{nl} \sum_{j=i}^{nl} G_{ij} [V_i^2 + V_j^2 - 2V_i V_j \cos(\delta_i - \delta_j)]$$

where $V_i$ is the voltage magnitude at the $i$th bus. $G_{ij}$ denotes the transfer conductance between bus $i$ and $j$, and $\delta_i$ is the voltage angle at the $i$th bus.

2) MINIMIZATION OF THE VOLTAGE DEVIATION
The voltage deviation is calculated as the sum of all load bus voltages deviating in the network from the ideal value by 1.0 p.u. Mathematically, the voltage deviation objective function ($F_{\text{VD}}$) can be defined as [20]:

$$F_{\text{VD}} = \left( \sum_{m=1}^{NL} |V_{L_m} - 1| \right)$$

B. CONSTRAINTS
1) EQUALITY CONSTRAINTS
The power balance equations are stated as follows:

$$P_{Gi} - P_{Di} - V_i \sum_{j=1}^{NB} V_j [G_{ij} \cos(\delta_i - \delta_j) + B_{ij} \sin(\delta_i - \delta_j)] = 0$$

$$Q_{Gi} - Q_{Di} - V_i \sum_{j=1}^{NB} V_j [G_{ij} \sin(\delta_i - \delta_j) - B_{ij} \cos(\delta_i - \delta_j)] = 0$$

where $P_G$ and $Q_G$ are the active and reactive power of the generator, respectively. $P_D$ and $Q_D$ represent the active and reactive load demands, respectively. $B_{ij}$ denotes the susceptance between bus $i$ and $j$, and $NB$ is the number of buses.

2) INEQUALITY CONSTRAINTS
To ensure a safe and stable operation, control variables and state variables necessarily satisfy the following inequality constraints.

$$V_{G_i}^{\text{min}} \leq V_{G_i} \leq V_{G_i}^{\text{max}}, \quad \forall i \in NG$$

$$Q_{C_j}^{\text{min}} \leq Q_{C_j} \leq Q_{C_j}^{\text{max}}, \quad \forall j \in NC$$

$$T_k^{\text{min}} \leq T_k \leq T_k^{\text{max}}, \quad \forall k \in NT$$
\[
\begin{aligned}
V_{l_{m}}^\text{min} & \leq V_{l_{m}} \leq V_{l_{m}}^\text{max}, \quad \forall m \in \text{NL} \\
G_{G_{i}}^\text{min} & \leq G_{G_{i}} \leq G_{G_{i}}^\text{max}, \quad \forall i \in \text{NG} \\
S_{q} & \leq S_{q}^\text{max}, \quad \forall q \in n l
\end{aligned}
\]  

(11)

C. \( \varepsilon \)-CONSTRAINT HANDLING TECHNIQUE

The equality constraints in the ORPD problem can be converted into inequality constraints, and all constraints can be written as:

\[
G_{i}(x, u) = \begin{cases}
\max \{g_{i}(x, u), 0\} & i = 1, 2, \ldots, m \\
\max \{|h_{i}(x, u)| - \delta, 0\} & i = m + 1, \ldots, p
\end{cases}
\]

where \( \delta \) signifies the tolerance parameter. Therefore, the individual’s total constraint violation can be formulated as:

\[
\text{vio}(x, u) = \frac{1}{\sum_{i=1}^{m} w_{i}} \cdot G_{i}(x, u)
\]

(13)

\[
w_{i} = \frac{1}{G_{i_{\text{max}}}}
\]

(14)

where \( w_{i} \) is a weight parameter and \( G_{i_{\text{max}}} \) represents the maximum violation of the constraint \( G_{i}(x, u) \) obtained so far.

The \( \varepsilon \)-constraint handling method is an effective constraint handling technique, which is first introduced by Takahama and Sakai [18]. The core idea is to find feasible solutions by controlling the \( \varepsilon \)-parameter [11]. The traditional \( \varepsilon \) setting method is stated as follows:

\[
\varepsilon (1) = \text{vio}(x^{\theta})
\]

(15)

\[
\varepsilon (t) = \begin{cases}
\varepsilon (1) \times \left(1 - \frac{t}{T_{c}}\right)^{\exp}, & 0 < t < T_{c} \\
0, & t < T_{c}
\end{cases}
\]

(16)

where \( x^{\theta} \) is the \( \theta \)th individual with the highest overall constraint violation in the initial population and \( \theta = 0.05 \times N \). \( \varepsilon (1) \) is the initial epsilon parameter and \( t \) indicates the current iteration. \( T_{c} \) represents the number of generations that the \( \varepsilon \)-parameter is managed before being set to zero. According to [18], the relevant parameters are set as \( cp \in [2, 10], T_{c} \in [0.1 \times t_{\text{max}}, 0.8 \times t_{\text{max}}] \), where \( t_{\text{max}} \) is the maximum number of iterations. We can follow the feasibility rule proposed by Deb to find an optimal feasible solution [21].

III. THE UNCERTAINTIES OF LOAD DEMAND AND RERs

A. UNCERTAINTY MODELING OF SOLAR IRRADIANCE

The lognormal PDF is applied to represent the solar irradiance [11] and expressed as follows:

\[
f_{G}(G) = \frac{1}{G \sigma_{s} \sqrt{2\pi}} \exp \left[ -\frac{\ln(G - \mu_{s})^{2}}{2\sigma_{s}^{2}} \right] \text{ for } G > 0
\]

(17)

where \( \mu_{s} \) and \( \sigma_{s} \) denote the mean and standard deviation, respectively. In this paper, the 1000 Monte Carlo scenarios of solar irradiance are created using lognormal PDF (\( \mu_{s} = 5.5, \sigma_{s} = 0.5 \)).

The output power of the photovoltaic unit as a function of irradiance can be calculated as [22]:

\[
P_{s}(G) = \begin{cases}
P_{sr} \left( \frac{G^{2}}{G_{sd} \times X_{c}} \right) & \text{for } 0 < G \leq X_{c} \\
P_{sr} \left( \frac{G}{G_{sd}} \right) & \text{for } G \geq X_{c}
\end{cases}
\]

(18)

where \( P_{sr} \) is the photovoltaic unit rated power which equals 50 MW; \( G_{sd} \) and \( X_{c} \) denote the standard solar irradiance and a certain irradiation point which are equal to 1000W/m\(^2\) and 120W/m\(^2\), respectively [11].

B. UNCERTAINTY MODELING OF LOAD

The uncertainty of load is modeled using the normal PDF defined as follows [23]:

\[
f_{d}(P_{d}) = \frac{1}{\sigma_{d} \sqrt{2\pi}} \exp\left[ -\frac{(P_{d} - \mu_{d})^{2}}{2\sigma_{d}^{2}} \right]
\]

(19)

where \( P_{d} \) represents probability density of normal distribution of load; \( \sigma_{d} \) and \( \mu_{d} \) are the standard deviation and mean values which are equal to 10 and 70, respectively [11]. Here, the Monte Carlo method is used to establish load demand scenarios (sample size 1000).

C. UNCERTAINTY MODELING OF WIND SPEED

The Weibull PDF is utilized to signify the wind speed [11] and is expressed as follows:

\[
f_{w}(v_{w}) = \left( \frac{\beta}{\alpha} \right)^{(\beta - 1)} \frac{v_{w}^{(\beta - 1)}}{\alpha^{\beta}} \exp\left[ -\frac{v_{w}}{\alpha}^{(\beta - 1)} \right] \quad 0 \leq v < \infty
\]

(20)

where \( v_{w} \) represents the wind speed, and its unit is meters per second. \( \alpha \) and \( \beta \) are the scale and shape parameters of the Weibull PDF, respectively. Weibull PDF is employed in the Monte Carlo method to construct the wind speed distribution scenarios (sample size 1000, \( \alpha = 9, \beta = 2 \)).

The output power of a wind turbine can be formulated as follows [24]:

\[
P_{w}(v_{w}) = \begin{cases}
0 & \text{for } v_{w} < v_{wr} \text{ and } v_{w} > v_{wo} \\
P_{or} \left( \frac{v_{w} - v_{wl}}{v_{wr} - v_{wl}} \right) & \text{for } v_{wl} \leq v_{w} \leq v_{wr} \\
P_{or} & \text{for } v_{wr} \leq v_{w} \leq v_{wo}
\end{cases}
\]

(21)

where \( P_{or} \) is the wind turbine rated output power; \( v_{wr} = 16 \text{ m/s}, v_{wl} = 3 \text{ m/s}, \) and \( v_{wo} = 25 \text{ m/s} \) are the rated, cut-in, and cut-out speeds of the wind turbine, respectively. Note that a wind farm is made up of 25 turbines with a total output of 75 MW.

D. BACKWARD REDUCTION TECHNIQUE

In this paper, the backward reduction algorithm [25] is applied to simplify 1000 Monte Carlo scenarios into 25 representative scenarios sets. Detailed steps of scenario reduction are described in [11]. The selected scenarios, along with
their associated parameters and probabilities, are provided in Table 3.

**IV. BACKGROUND**

**A. MARINE PREDATORS ALGORITHM**

MPA is a novel nature-inspired metaheuristic algorithm that simulates the biological behavior of marine predators foraging for prey [14]. The main operation idea is that predators adopt two random walks called Lévy flight and Brownian motion to improve the encounter rate with prey and update their position based on the optimal solution. Meanwhile, the prey also acts as a predator. Because the prey is hunting for its own food while the predator is looking for prey.

Similar to other population-based metaheuristic algorithms, the MPA also randomly initializes the location of prey through (22):

\[ \overrightarrow{\mathcal{X}}_i = \overrightarrow{\mathcal{b}} + \text{rand} \otimes (\overrightarrow{\mathcal{u}b} - \overrightarrow{\mathcal{b}}), \quad i = 1, 2, \ldots, n \]  

(22)

where \( \overrightarrow{\mathcal{u}b} \) and \( \overrightarrow{\mathcal{b}} \) are the upper and lower boundaries of the search space, respectively. Also, \( \text{rand} \) represents a random number within the range from 0 to 1. The prey matrix is described as follows:

\[ \text{Prey} = \begin{bmatrix} X_{1,1} & X_{1,2} & \ldots & X_{1,d} \\ X_{2,1} & X_{2,2} & \ldots & X_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \ldots & X_{n,d} \end{bmatrix} \]  

(23)

where \( n \) is the population size and \( d \) signifies problem dimensions.

In this step, according to the fitness value of prey, we determine the best solution as the top predator, namely the Elite matrix, which is defined as (24):

\[ \text{Elite} = \begin{bmatrix} X'_{1,1} & X'_{1,2} & \ldots & X'_{1,d} \\ X'_{2,1} & X'_{2,2} & \ldots & X'_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ X'_{n,1} & X'_{n,2} & \ldots & X'_{n,d} \end{bmatrix} \]  

(24)

The entire iterative process is divided into three stages based on different speed ratios while simulating the lifespan of predators and prey. The description of these three stages are as follows [14]:

The first stage is carried out in the first third of the iteration when the prey is faster than the predator. The updated process for the prey is as follows:

\[
\overrightarrow{\text{Step}}_i = \overrightarrow{R}_B \otimes \left( \text{Elite}_i - \overrightarrow{R}_B \otimes \overrightarrow{\text{Prey}}_i \right), \quad i = 1, 2, \ldots, n/2
\]  

(25)

\[
\overrightarrow{\text{Prey}}_i = \overrightarrow{\text{Prey}}_i + \overrightarrow{R} \otimes \overrightarrow{\text{Step}}_i
\]  

(26)

where \( \overrightarrow{\text{Step}}_i \) is the step size vector, and \( \overrightarrow{R}_B \) is a normally distributed random vector, representing the Brownian motion. \( \overrightarrow{R} \in [0, 1] \) denotes a random vector, \( P = 0.5 \), and \( \otimes \) indicates the entry-wise multiplication. In addition, \( r \) and \( l_{max} \) are the present and maximum iterations, respectively.

The second stage is implemented in the middle of the iteration when the predator and prey move at the same speed. The population is split into two groups, one is used for exploration and the other is utilized for development. The mathematical descriptions of both are as follows:

\[
\overrightarrow{\text{Step}}_i = \overrightarrow{R}_L \left( \text{Elite}_i - \overrightarrow{R}_L \otimes \overrightarrow{\text{Prey}}_i \right), \quad i = 1, 2, \ldots, n/2
\]  

(27)

\[
\overrightarrow{\text{Prey}}_i = \overrightarrow{\text{Prey}}_i + \overrightarrow{P} \cdot \overrightarrow{R} \otimes \overrightarrow{\text{Step}}_i
\]  

(28)

where \( \overrightarrow{R}_L \) represents the random vector based on the Lévy distribution.

\[
\overrightarrow{\text{Step}}_i = \overrightarrow{R}_B \otimes \left( \overrightarrow{R}_B \otimes \text{Elite}_i - \overrightarrow{\text{Prey}}_i \right), \quad i = n/2, \ldots, n
\]  

(29)

\[
\overrightarrow{\text{Prey}}_i = \text{Elite}_i + \overrightarrow{P} \cdot \overrightarrow{CF} \otimes \overrightarrow{\text{Step}}_i
\]  

(30)

where \( \overrightarrow{CF} = \left( 1 - \frac{l}{l_{max}} \right)^{\frac{2.5}{l_{max}}} \) is an adaptive parameter to control the predator’s moving step.

The last stage is executed in the last third of the iteration when the predator outruns the prey. The updated formula for the prey is as follows:

\[
\overrightarrow{\text{Step}}_i = \overrightarrow{R}_L \otimes \left( \overrightarrow{R}_L \otimes \text{Elite}_i - \overrightarrow{\text{Prey}}_i \right)
\]  

(31)

\[
\overrightarrow{\text{Prey}}_i = \text{Elite}_i + \overrightarrow{P} \cdot \overrightarrow{R} \otimes \overrightarrow{\text{Step}}_i
\]  

(32)

However, some studies have shown that eddy formation or the effect of fish aggregating devices (FADs) would affect the foraging behavior of predators, and easily lead to local optimum. This process can be mathematically described as in (33), shown at the bottom of the page, where \( \overrightarrow{U} \) is a binary vector, \( \text{FADs} = 0.2 \), and \( r \in [0, 1] \). \( r_1 \) and \( r_2 \) are random indices from the prey matrix.

**B. SPHERICAL SEARCH ALGORITHM**

Spherical search algorithm (SS) is proposed by Abhishek Kumar to solve global optimization problems with bounded constraints [26]. In the SS algorithm, each solution creates its own food while the predator is looking for prey. Because the prey is hunting for its own food while the predator is looking for prey.

The effect of fish aggregating devices (FADs) would affect the foraging behavior of predators, and easily lead to local optimum. This process can be mathematically described as in (33), shown at the bottom of the page, where \( \overrightarrow{U} \) is a binary vector, \( \text{FADs} = 0.2 \), and \( r \in [0, 1] \). \( r_1 \) and \( r_2 \) are random indices from the prey matrix.

\[
\overrightarrow{\text{Prey}}_i = \begin{cases} \overrightarrow{\text{Prey}}_i + \overrightarrow{P} \cdot \overrightarrow{R} \otimes \overrightarrow{\text{Step}}_i & \text{if } r < \text{FADs} \\ \overrightarrow{\text{Prey}}_i + \left[ \text{FADs} \left( 1 - r \right) + r \right] \left( \overrightarrow{\text{Prey}}_i - \overrightarrow{\text{Prey}}_{i-1} \right) & \text{otherwise} \end{cases}
\]  

(33)
Similar to other metaheuristic algorithms, the SS algorithm initializes the search process by randomly generating a set of populations, and the expression is shown in (22). Then the ith trial solution \( \overrightarrow{Y}_i \) corresponding to the ith solution \( \overrightarrow{X}_i \) can be calculated as:

\[
\overrightarrow{Y}_i = \overrightarrow{X}_i + c_i Pr_i \overrightarrow{Z}_i
\]  
(34)

where \( c_i \) represents the step-size control vector of the ith trial solution, whose value is randomly calculated within the range of \([0.5, 0.7]\) obtained from the experiment. \( Pr_i \) is the projection matrix, which determines the value of \( \overrightarrow{Y}_i \) on the spherical boundary, and \( \overrightarrow{Z}_i \) denotes the search direction.

Algorithm 1 Pseudocode of the SS

**Input:** \( n, NE_{\text{max}}, ub, lb, rd = 0.95, c = [0.5, 0.7] \)

\( n \) is population size, \( NE_{\text{max}} \) is the maximum number of calculations, \( ub \) and \( lb \) are the boundaries of the search space, \( d \) is search-space dimension; \( c \) is the parameter for the scale of step-size w/

**Output:** The best solution

1. Initialization
2. Calculate the fitness value of \( \overrightarrow{X}_i \)
3. While \( (FEs \leq FE_{\text{max}}) \)
4. Calculate step-size control vector, \( c_i \)
5. Calculate the search direction of \( \overrightarrow{X}_i \)
6. Calculate the orthogonal matrix, \( A \)
7. for \( i = 1 \) to \( n \)
8. Calculate the rank of projection using (38)
9. if \( i < 0.5 * n \) then
10. Update the half population of better solutions using (35)
11. else
12. Update the other half of the population using (36)
13. endif
14. end for
15. Calculate the trial solution
16. Apply the greedy selection
17. end while

The SS algorithm uses two ways to calculate the search direction, namely toward-rand and toward-best. The population is divided into two groups on average according to the fitness value. When determining the search direction, the group with superior fitness uses toward-rand, while the group with poor fitness uses toward-best. The mathematical descriptions of both are as follows:

\[
\overrightarrow{Z}_i = \overrightarrow{X}_{pi} + \overrightarrow{X}_{qi} - \overrightarrow{X}_{ri} - \overrightarrow{X}_i
\]  
(35)

\[
\overrightarrow{Z}_i = \overrightarrow{X}_{pbest} + \overrightarrow{X}_{qi} - \overrightarrow{X}_{ri} - \overrightarrow{X}_i
\]  
(36)

where \( pi, qi \) and \( ri \) are randomly selected indices from among 1 to \( n \) such that \( pi \neq qi \neq ri \neq p \). \( pbest \) denotes the randomly selected individuals from the top \( p \) solutions in the current iteration.

The projection matrix \( Pr_i \) in (38) is a symmetric matrix, and its expression is as follows:

\[
Pr = A' \text{diag} \left( \overrightarrow{b}_i \right) A
\]  
(37)

where \( A \) is an orthogonal matrix and \( \overrightarrow{b}_i \) represents a binary vector. The calculation method of binary diagonal matrix \( \text{diag} \left( \overrightarrow{b}_i \right) \) is as follows:

\[
0 < \text{rank} \left( \text{diag} \left( \overrightarrow{b}_i \right) \right) < d
\]  
(38)

Followed by using greedy selection to update the ith solution of the population, the mathematical expression can be expressed as follows:

\[
\overrightarrow{X}_{i+1} = \begin{cases} 
\overrightarrow{Y}_i, & \text{if } f(\overrightarrow{Y}_i) \leq f(\overrightarrow{X}_i) \\
\overrightarrow{X}_i, & \text{otherwise}
\end{cases}
\]  
(39)

The basic steps of the SS algorithm are summarized by the pseudo-code shown in Algorithm 1.

V. HYBRID IMPROVED MARINE PREDATORS ALGORITHM

A. OPPOSITION-BASED LEARNING

Opposition-based learning (OBL) is an effective method of population initialization proposed by Tizhoosh [27]. In the metaheuristic algorithm, some randomly generated search agents are frequently distributed far away from the optimal solution or in the opposite position of the optimal solution, which reduces the search efficiency of the population. Here, the OBL strategy improves the quality of the initial population by constructing a solution closer to the optimal solution search region. Suppose \( \overrightarrow{X}_i = (X_{i,1}, X_{i,2}, \ldots, X_{i,d}) \) is a solution in \( d \)-dimensional space, then the opposite solution \( \overrightarrow{X}'_i \) is calculated as follows:

\[
\overrightarrow{X}'_i = \overrightarrow{ub} - \overrightarrow{lb} - \overrightarrow{X}_i, \quad i = 1, 2, \ldots, n
\]  
(40)

Therefore, the fittest search agents are chosen from the population generated by random and its opposite population to construct an initial population.

B. TRANSITIONAL STAGE OF INTEGRATING DIFFERENTIAL EVOLUTION

In the preliminary and intermediate stages of the MPA, predators and prey may skip the most promising area in the search area because they search for food quickly [28]. Inspired by the DE algorithm, the improved DE/best – to-rand/1 strategy is used to replace the middle stage of the traditional MPA, which balances the development and exploration capabilities and improves the convergence speed of the algorithm. The updated mode for the prey position is as follows:

\[
\overrightarrow{Prey}_{i} = \overrightarrow{Elite}_{i} + \text{rand} \cdot \left( \overrightarrow{Prey}_{i} - \overrightarrow{Prey}_{i+1} \right) + \text{rand} \cdot \left( \overrightarrow{Prey}_{i+5} - \overrightarrow{Prey}_{i+6} \right)
\]  
(41)
where $\text{rand}$ represents a random value within [0, 1]. The subscripts $r_4$, $r_5$ and $r_6$ indicate that integers are randomly selected from 1 to $n$ with $r_4 \neq r_5 \neq r_6 \neq i$. Note that this phase is executed between $t > 1/20 \max$ and $t < 18/20 \max$.

C. ADAPTIVE MUTATION OPERATION
In the third stage of iterative optimization, the SS operator is used to strengthen the development stage of the traditional MPA, to improve the search efficiency of the improved algorithm. Here, the probability value of $p = 0.7$ is utilized to perform the adaptive mutation operation, and the prey updates their position by performing MPA or SS operators to exploit the finest solutions. The corresponding mathematical expression of this strategy is:

$$\overrightarrow{\text{Prey}}_i = \begin{cases} \text{operators of MPA} & \text{rand} < p \\ \text{operators of SS} & \text{otherwise} \end{cases} \quad (42)$$

Algorithm 2 Pseudocode of the HIMPA

**Input:** $n, \max, ub, lb, P = 0.5, \text{FADs} = 0.2$

1. Population size, $\max$ is the maximum number of iterations, $ub$ and $lb$ are the boundaries of the search space \*$/.

2. Top predator
3. Initialize search agents using the OBL strategy
4. $t = 1$
5. while ($t \leq \max$)
6. Calculate fitness and constraint violation for each search agent
7. Construct the Elite matrix
8. Accomplish memory saving according to constraint violation
9. \textbf{if} $t < \max/20$ \textbf{then}
10. Update prey using ((25)-(26))
11. $t_{\max}/20 < t < 18 \times \max/20$ \textbf{then}
12. Update prey using (41)
13. \textbf{else}
14. Update prey using (42)
15. end if
16. Update $\epsilon$-parameter, evaluate fitness and constraint violation for each search agent
17. Accomplish memory saving according to the rules of EC and Elite individual update
18. Apply the FADs effect using (33), and $t = t + 1$
19. end while

D. THE FRAMEWORK OF HIMPA
The basic steps of the proposed HIMPA are summarized by the pseudo-code shown in Algorithm 2.

VI. SIMULATION RESULTS AND DISCUSSION
In this section, the HIMPA is studied for solving the ORPD problem on IEEE 30-bus systems with and without RERs uncertainty. The parameters and generator data of the studied system are taken from [11]. For all given cases, the population size, the number of trial runs, the maximum number of function evaluations, and the maximum number of iterations are specified as 40, 5, 30000, and 374, respectively. The proposed HIMPA runs on MATLAB R2020a with an Intel Core i3 @4 GHz CPU with 64GB RAM.

A. CASE 1: STANDARD IEEE 30-BUS SYSTEM
Cases (1) and (2) aim to minimize real power loss ($P_{\text{loss}}$) and aggregate voltage deviation ($VD$), respectively. In both cases, the shunt compensator and transformer taps are set to discrete variables in accordance with the recommendations in [11]. In addition, the active power value of generators should be reasonably selected within the specified range of the generators. To effectively compare with previous studies, the objective functions of Cases (1a) and (2a) are to reduce $P_{\text{loss}}$ and $VD$, respectively. And these cases take into account the active power value of the generator.

Table 1 provides the optimization results of all variables of the four study cases. In Case (1), the $P_{\text{loss}}$ obtained by the HIMPA is 4.4125 MW, while the $VD$ in Case (2) is 0.08846 p.u. The voltage profiles of load buses are shown in Fig. 1. Voltage of few load buses operates near the maximum limit to achieve the minimum $P_{\text{loss}}$. While Cases (2) and (2a) will not result in excess voltage because the goal is to keep the bus voltage close to 1.0 p.u.

Figs. 2 and 3 show the iterative convergence curves of the HIMPA and MPA-EC in Cases (1) and (2), respectively. It can be seen that the MPA-EC is liable to fall into local optimization in the early and middle stages of optimization, and the optimal solution cannot be found quickly. However, the HIMPA can converge to the optimal value quickly in the early stage of optimization. This shows that the improved marine predator algorithm effectively balances the global and local search ability. Table 2 presents a comparison between the proposed algorithm in this study and the reported algorithm. None of the other references in the table specifies the active power settings for the particular generators, except [29] and [30]. Assume that the data in these references are set according to the data in [29]. The study in [11] analyzes in detail the infeasibility of some algorithm solutions in Table 2. In the comparison between Case (2) and (2a), many reported algorithms have better $VD$ values than the proposed HIMPA, but there is no documented study to examine the corresponding actual reactive power generation status or voltage distribution of load buses [11].

B. CASE 2: MODIFIED IEEE 30-BUS SYSTEM
This subsection adopts the stochastic ORPD approach based on wind power uncertainty proposed in [31] and [32] to calculate the expected power loss ($EPL$) and expected voltage deviation ($EVD$). When RERs are in short supply, swing generators necessarily provide additional power to meet system demand. The details of stochastic ORPD are explained in [16]. The modified IEEE 30-bus system is to switch the...
TABLE 1. Simulation results of the study cases.

| Parameters | Min | Max | Case 1 (Min, \(P_{\text{loss}}\)) | Case 1a (Min, \(P_{\text{loss}}\)) | Case 2 (Min, \(V_D\)) | Case 2a (Min, \(V_D\)) |
|------------|-----|-----|----------------------------------|----------------------------------|------------------------|------------------------|
| \(V_1\) (p.u.) | 0.95 | 1.1 | 1.0695                           | 1.0717                           | 1.0012                 | 1.0032                 |
| \(V_2\) (p.u.) | 0.95 | 1.1 | 1.0608                           | 1.0626                           | 0.9978                 | 0.9986                 |
| \(V_7\) (p.u.) | 0.95 | 1.1 | 1.0372                           | 1.0403                           | 1.0173                 | 1.0164                 |
| \(V_8\) (p.u.) | 0.95 | 1.1 | 1.0434                           | 1.0404                           | 1.0072                 | 1.0083                 |
| \(V_{11}\) (p.u.) | 0.95 | 1.1 | 1.08                            | 1.0928                           | 1.0599                 | 1.0726                 |
| \(V_{13}\) (p.u.) | 0.95 | 1.1 | 1.0488                           | 1.057                            | 1.0211                 | 1.0161                 |
| \(T_{11}\) (p.u.) | 0.9 | 1.1 | 1.1                            | 1.0646                           | 1.08                     | 1.0977                 |
| \(T_{12}\) (p.u.) | 0.9 | 1.1 | 0.9                            | 0.914                            | 0.9                     | 0.9007                 |
| \(T_{15}\) (p.u.) | 0.9 | 1.1 | 0.98                           | 0.9887                           | 1                      | 0.9959                 |
| \(T_{36}\) (p.u.) | 0.9 | 1.1 | 0.98                           | 0.9753                           | 0.96                     | 0.9696                 |
| \(Q_{C1}\) (MVar) | 0 | 5 | 5                            | 0.028                            | 4.6                     | 4.995                  |
| \(Q_{C2}\) (MVar) | 0 | 5 | 5                            | 0.2835                           | 0.6                     | 3.6847                 |
| \(Q_{C3}\) (MVar) | 0 | 5 | 4.4                           | 4.2083                           | 5                      | 4.992                  |
| \(Q_{C4}\) (MVar) | 0 | 5 | 5                            | 4.9874                           | 0                      | 0.0363                 |
| \(Q_{C5}\) (MVar) | 0 | 5 | 3.8                           | 3.9949                           | 5                      | 5                      |
| \(Q_{C6}\) (MVar) | 0 | 5 | 5                            | 4.9968                           | 5                      | 4.9903                 |
| \(Q_{C7}\) (MVar) | 0 | 5 | 5                            | 2.8392                           | 5                      | 4.9945                 |
| \(Q_{C8}\) (MVar) | 0 | 5 | 2.6                           | 4.9991                           | 5                      | 5                      |
| \(R_{loss}\) (MW) | 4.4125 | 4.8613 | 5.0567 | 6.0115 | 0.08846 | 0.08766 |
| \(V_D\) (p.u.) | 0.8827 | 0.9203 | 0.9203 | 0.8528 | 0.8528 | 0.8528 |
| \(Q_{G1}\) (MVar) | 0 | 5 | 5                            | 0.0571                           | 0.0506                 | 19.9681                |
| \(Q_{G2}\) (MVar) | 0 | 5 | 5                            | 15.4242                          | 16.0409                | 17.8474                |
| \(Q_{G3}\) (MVar) | 0 | 5 | 24.7793                        | 24.4071                          | 59.4509                | 53.7128                |
| \(Q_{G4}\) (MVar) | 0 | 5 | 29.0865                        | 29.1949                          | 45.5358                | 48.6412                |
| \(Q_{G5}\) (MVar) | 0 | 5 | 27.2883                        | 27.7897                          | 31.1177                | 37.8465                |
| \(Q_{G6}\) (MVar) | 0 | 5 | 44.7                          | 0.3521                           | 5.5709                 | 5.5332                 |

FIGURE 1. Voltage profiles of load buses in different cases.

FIGURE 2. Convergence characteristics of HIMPA and MPA-EC for Case 1.

thermal generators of buses 5 and 8 to wind generator and solar generator units, respectively. Table 3 lists the details of 25 representative scenarios. The objective of Case (3) is to minimize the \(EPL\) under uncertain conditions, as follows:

\[
EPL = \sum_{SC=1}^{N_{SC}} \Delta_{SC} \times P_{\text{loss,SC}} \quad (43)
\]

where \(\Delta_{SC}\) and \(P_{\text{loss}}\) represent the probability and minimum power loss of the scenario, respectively. Similarly, the minimization of the \(EVD\) under all scenarios is evaluated as follows:

\[
EVD = \sum_{SC=1}^{N_{SC}} \Delta_{SC} \times VD_{SC} \quad (44)
\]
The EPL value of Case (3) is 1.8316 MW, and the EVD value of Case (4) is 0.05856 p.u. According to the data analysis in Table 3, under the condition of minimum network load, $P_{loss}$ is also small (scenario 4). Minimum loading represents the lowest current in the network and therefore the lowest power loss. On the other hand, when the network loading is the largest, the system active power loss is the largest (scenario 20).

The experimental results of the EVD show that under the minimum load (scenario 4), the bus voltage levels of the network can be maintained close to the desired 1.0 p.u. Similarly, the low current means the minimum aggregate $VD$. 

![Figure 3](image1.png)  
**FIGURE 3.** Convergence characteristics of HIMPA and MPA-EC for Case 2.

![Figure 4](image2.png)  
**FIGURE 4.** Voltage profiles of load buses in extreme scenarios.

![Figure 5](image3.png)  
**FIGURE 5.** Optimum values of generator bus voltages in Case 3.

![Figure 6](image4.png)  
**FIGURE 6.** Optimum values of generator bus voltages in Case 4.

**TABLE 2.** Comparison results of different cases.

| Case no. | Algorithm | $P_{loss}$ (MW) | $VD$ (p.u.) |
|----------|-----------|-----------------|-------------|
| Case 1   | HIMPA     | 4.4125          | 0.8827      |
|          | MPA-EC    | 4.4348          | 0.9368      |
|          | SHADE-EC [11] | 4.4126    | 0.9029      |
|          | AEFA [30] | 4.4711          | 0.8839      |
|          | JS [30]   | 4.4289          | 0.8564      |
| Case 1a  | HIMPA     | 4.8613          | 0.9203      |
|          | MPA-EC    | 4.888           | 0.8767      |
|          | SHADE-EC [11] | 4.8612    | 0.9205      |
|          | AEFA [30] | 4.9393          | 0.8938      |
|          | JS [30]   | 4.8635          | 0.9207      |
|          | DE [5]    | 4.5550          | 1.9589<sup>a</sup> |
|          | HFA [33]  | 4.529           | 1.625<sup>b</sup> |
|          | IGSA-CSS [34] | 4.7660     | -           |
|          | NGBWCA [35] | 4.4801    | 0.8413      |
|          | QOTLBO [36] | 4.5594    | 1.9057<sup>b</sup> |
|          | EMA [29]  | 4.4798          | 0.8123      |
|          | FAHCLPSO [37] | 4.4877    | -           |
|          | OGSA [38] | 4.4984          | 0.8085      |
|          | MFO [39]  | 4.5128          | -           |
| Case 2   | HIMPA     | 5.5967          | 0.08846     |
|          | MPA-EC    | 5.5356          | 0.1227      |
|          | SHADE-EC [11] | 5.4495    | 0.08886     |
|          | AEFA [30] | 5.3886          | 0.1418      |
|          | JS [30]   | 5.3390          | 0.0947      |
| Case 2a  | HIMPA     | 6.0115          | 0.08766     |
|          | MPA-EC    | 5.8497          | 0.09687     |
|          | SHADE-EC [11] | 6.0999    | 0.08724     |
|          | AEFA [30] | 5.7490          | 0.1313      |
|          | JS [30]   | 5.8489          | 0.0944      |
|          | HFA [33]  | 5.75           | 0.0980      |
|          | IGSA-CSS [34] | -        | 0.08968<sup>a</sup> |
|          | NGBWCA [35] | 6.3142    | 0.0458      |
|          | QOTLBO [36] | 6.4962    | 0.0856      |
|          | EMA [29]  | 6.241          | 0.06131     |
|          | FAHCLPSO [37] | -        | 0.04315     |
|          | OGSA [38] | 6.9044          | 0.0640      |
|          | MFO [39]  | -              | 0.12154     |

<sup>a</sup> The range of load bus voltage and generator reactive power are relaxed.
<sup>b</sup> Infeasible method, load bus voltage regulations are violated.
On the contrary, under the maximum load (scenario 20), the aggregate VD of the system is the highest.

Fig. 4 depicts the distribution of load bus voltage in these two extreme scenarios. Figs. 5 and 6 show the optimal values of generator bus voltages for each scenario in Cases (3) and (4), respectively. Generator bus voltage values in Case (3) are higher than those in Case (4); particularly in Case (4), bus 11 has a greater voltage setting, since it is connected to the nearby load buses without any shunt compensators.

VII. CONCLUSION

In this paper, a hybrid improved marine predators algorithm (HIMPA) is proposed, which can improve the performance of the original MPA when solving the ORPD problem. The three improvements of the HIMPA are based on opposition-based learning, the transitional stage of integrating differential evolution, and the adaptive mutation strategy. In the first section, the proposed HIMPA is applied to solve the deterministic ORPD problem, and the basic IEEE 30-bus system is selected for testing. The optimization objectives are real power loss and voltage deviation. The test results show that the optimal solution obtained by HIMPA is better than other competitive algorithms. Subsequently, stochastic ORPD solutions with uncertainties of solar power, wind, and load demand have been investigated by a scenario-based approach for the modified IEEE 30-bus system. Various scenarios are created by Monte Carlo simulation, and then a group of representative scenarios is selected by using the backward reduction technique. The expected power loss and expected voltage deviation values in different scenarios are calculated by optimizing these parameters. The results show that the combination of the HIMPA and ε-constraint technique can effectively solve the ORPD problem.

**REFERENCES**

[1] K. B. O. Medani, S. Sayah, and A. Bekrar, “Whale optimization algorithm based optimal reactive power dispatch: A case study of the Algerian power system,” Electr. Power Syst. Res., vol. 163, pp. 696–705, Oct. 2018.
[2] T. T. Nguyen, D. N. Vo, H. Van Tran, and L. Van Dai, “Optimal dispatch of reactive power using modified stochastic fractal search algorithm,” Complexity, vol. 2019, May, Art. no. 4670820.
[3] X. Liu, “Optimal reactive power dispatch with discrete variables,” J. Comput. Theor. Nanosci., vol. 13, no. 3, pp. 1913–1919, Mar. 2016.
[4] Z. Zhao, S. Liu, M. Zhou, and A. Abusorrah, “Dual-objective mixed integer linear program and memetic algorithm for a group-scheduling problem,” IEEE/CAA J. Autom. Sinica, vol. 8, no. 6, pp. 1199–1209, Jun. 2021.
[5] A. A. A. E. Ela, M. A. Abido, and S. R. Spea, “Differential evolution algorithm for optimal reactive power dispatch,” Electr. Power Syst. Res., vol. 81, no. 2, pp. 458–464, Feb. 2011.
[6] R. Jamal, B. Men, and N. H. Khan, “A novel nature inspired meta-heuristic optimization approach of GO-based optimizer for optimal reactive power dispatch problems,” IEEE Access, vol. 8, pp. 202596–202610, 2020.
[7] J. Zhao, S. Liu, M. Zhou, X. Guo, and L. Qi, “Modified cuckoo search algorithm to solve economic power dispatch optimization problems,” IEEE/CAA J. Autom. Sinica, vol. 7, no. 4, pp. 794–806, Jul. 2020.
[8] S. Mugemanyi, Z. Qu, F. X. Rugema, Y. Dong, C. Bananeza, and L. Wang, “Optimal reactive power dispatch using chaotic bat algorithm,” IEEE Access, vol. 8, pp. 65830–65867, 2020.
[9] T. T. Nguyen and D. N. Vo, “Improved social spider optimization algorithm for optimal reactive power dispatch problem with different objectives,” Neuro Comput. Appl., vol. 32, no. 10, pp. 5919–5950, May 2020.
[10] S. Kamel, S. Abedi-Fatah, M. Ebeed, J. Yu, K. Xie, and C. Zhao, “Solving optimal reactive power dispatch considering load uncertainty,” in Proc. IEEE Innov. Smart Grid Technol. Asia (ISGT Asia), May 2019, pp. 1335–1340.
[11] P. P. Biswas, P. N. Suganthan, R. Mallipeddi, and G. A. J. Amiratunga, “Optimal reactive power dispatch with uncertainties in load demand and renewable energy sources adopting scenario-based approach,” Appl. Soft Comput., vol. 75, pp. 616–632, Feb. 2019.
[12] M. Ebeed, A. Ali, M. I. Mosaad, and S. Kamel, “An improved lighting attachment procedure optimizer for optimal reactive power dispatch with uncertainty in renewable energy resources,” IEEE Access, vol. 8, pp. 168721–168731, 2020.
[13] A. Bazziar, R. Bo, M. D. Ghobabadi, M. Veisi, and W. Ur Rehman, “Evolutionary algorithm-based adaptive robust optimization for AC security constrained unit commitment considering renewable energy sources and shunt FACTS devices,” IEEE Access, vol. 9, pp. 123575–123587, 2021.
A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine predators algorithm: A nature-inspired metaheuristic,” Expert Syst. Appl., vol. 152, Aug. 2020, Art. no. 113377.

N. H. Khan, R. Jamal, M. Ebeed, S. Kamel, H. Zeinoddini-Meymand, and H. M. Zawbaa, “Adopting scenario-based approach to solve optimal reactive power dispatch problem with integration of wind and solar energy using improved marine predator algorithm,” Ain Shams Eng. J., vol. 13, no. 5, Sep. 2022, Art. no. 101726.

M. A. Elaziz, A. A. Ewees, D. Yousri, H. S. N. Alwerfali, Q. A. Awad, S. Lu, and M. A. A. Al-Qanes, “An improved marine predators algorithm with a fuzziness entropy for multi-level thresholding: Real world example of COVID-19 CT image segmentation,” IEEE Access, vol. 8, pp. 125306–125330, 2020.

M. A. Elaziz, S. B. Thanikanti, I. A. Ibrahim, S. Lu, B. Nastasi, M. A. Aloabi, M. A. Hossain, and D. Yousri, “Enhanced marine predators algorithm for identifying static and dynamic photovoltaic models parameters,” Energy Convers. Manag., vol. 236, May 2021, Art. no. 113971.

T. Takahama and S. Sakai, “Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites,” in Proc. IEEE Int. Conf. Evol. Comput., Jul. 2006, pp. 1–8.

J. Li, J. Zhou, and B. Chen, “Review of wind power scenario generation methods for optimal operation of renewable energy systems,” Appl. Energy, vol. 280, Dec. 2020, Art. no. 115992.

S. Li, W. Gong, L. Wang, X. Yan, and C. Hu, “Optimal power flow by means of improved adaptive differential evolution,” Energy, vol. 198, May 2020, Art. no. 117314.

K. Deb, “An efficient constraint handling method for genetic algorithms,” Comput. Methods Appl. Mech. Eng., vol. 186, nos. 2–4, pp. 311–338, Jun. 2000.

P. P. Biswas, P. N. Suganthan, and G. A. J. Amarangatunga, “Optimal power flow solutions incorporating stochastic wind and solar power,” Energy Convers. Manag., vol. 148, pp. 1194–1207, Sep. 2017.

J. Yang and C. Su, “Robust optimization of microgrid based on renewable distributed power generation and load demand uncertainty,” Energy, vol. 223, May 2021, Art. no. 120043.

J. Hetzer, D. C. Yu, and K. Bhattarai, “An economic dispatch model incorporating wind power,” IEEE Trans. Energy Convers., vol. 23, no. 2, pp. 603–611, Jun. 2008.

N. Grows-Kuska, H. Heitsch, and W. Romisch, “Scenario reduction and scenario tree construction for power management problems,” in Proc. Bologna Power Technol. Conf., vol. 3, Jun. 2003, p. 7.

A. Kumar, R. K. Misra, D. Singh, S. Mishra, and S. Das, “The spherical search algorithm for bound-constrained global optimization problems,” Appl. Soft Comput., vol. 85, Dec. 2019, Art. no. 105734.

H. R. Tizhoosh, “Opposition-based learning: A new scheme for machine intelligence,” in Proc. CIMCA-IWAITC, vol. 1, Vienna, Austria, 2005, pp. 695–701.

Q. Fan, H. Huang, Q. Chen, L. Yao, K. Yang, and D. Huang, “A modified self-adaptive marine predators algorithm: Framework and engineering applications,” Eng. Comput., vol. 38, no. 4, pp. 3269–3294, Mar. 2021.

A. Faramarzi, M. Heidarinejad, B. Mandal, and P. K. Roy, “Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization,” Int. J. Electr. Power Energy Syst., vol. 53, pp. 123–134, Dec. 2013.

E. Naderi, H. Narimani, M. Fathi, and M. R. Narimani, “A novel fuzzy adaptive configuration of particle swarm optimization to solve large-scale optimal reactive power dispatch,” Appl. Soft Comput., vol. 53, pp. 441–456, Apr. 2017.

B. Mandal and P. K. Roy, “Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization,” Int. J. Electr. Power Energy Syst., vol. 53, pp. 123–134, Dec. 2013.

N. Growe-Kuska, H. Heitsch, and W. Romisch, “Scenario reduction and constraint handling by improved GSA-based algorithm with the novel strategies to handle constraints,” Optimization, vol. 66, nos. 1–2, pp. 179–198, Jan. 2017.

A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine predators algorithm: A nature-inspired metaheuristic,” Expert Syst. Appl., vol. 152, Aug. 2020, Art. no. 113377.