

A novel hybrid approach based artificial bee colony and salp swarm algorithms for solving ORPD problem

Souheil Salhi¹, Djemai Naimi², Ahmed Salhi³, Saleh Abujarad⁴, Abdelouahab Necira⁵

^{1,2,3,5}Laboratory of Electrical Engineering, Department of Electrical Engineering, University Mohamed khider of Biskra, Biskra, Algeria

⁴Department of Electromechanical, Systems and Metal Engineering, Ghent University, Ghent, Belgium

Article Info

Article history:

Received May 28, 2021

Revised Jul 27, 2021

Accepted Aug 6, 2021

Keywords:

Hybrid ABC and SSA algorithm

ORPD

Total transmission active losses

TVD

VSI

ABSTRACT

Optimal reactive power dispatch (ORPD) is an important task for achieving more economical, secure and stable state of the electrical power system. It is expressed as a complex optimization problem where many meta-heuristic techniques have been proposed to overcome various complexities in solving ORPD problem. A meta-heuristic search mechanism is characterized by exploration and exploitation of the search space. The balance between these two characteristics is a challenging problem to attain the best solution quality. The artificial bee colony (ABC) algorithm as a reputed meta-heuristic has proved its goodness at exploration and weakness at exploitation where the enhancement of the basic ABC version becomes necessary. Salp swarm algorithm (SSA) is a newly developed swarm-based meta-heuristic, which has the best local search capability by using the best global solution in each iteration to discover promising solutions. In this paper, a novel hybrid approach-based ABC and SSA algorithms (ABC-SSA) is that developed to enhance the exploitation capability of the ABC algorithm using SSA and applied for solving ORPD problem. The efficiency of ABC-SSA is investigated using two standard test systems IEEE-30 and IEEE-300 buses, and that by considering the famous objective functions in ORPD problem.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Souheil Salhi

Department of Electrical Engineering

University Mohamed khider of Biskra

BP 145, 07000 Biskra, Algeria

Email: s.souhail@univ-biskra.dz

NOMENCLATURE

P_G / Q_G	: Active power generation/reactive power generator
P_l / Q_l	: Active power of load demand/reactive power of load demand
S_l	: Apparent power flow of branch l -th
G_k	: Conductance of k -th branch of the network
δ_{ij}	: Difference of voltage angle between buses i and j
Q_c	: Injected VAR power from compensator
I_{PV} / I_{PQ}	: Injected current vector of generator and load buses, respectively
$F(x,u)$: Objective function of the problem.
Y_{ij} / α_{ij}	: Magnitude/angle of admittance matrix element between buses i and j
NLB / NG	: Number of load buses/number of generator buses
NTL / NT	: Number of transmission lines/number of tap setting transformers

NC/NB	: Number of shunt capacitor banks/number of total buses
T_k	: Ratio of k -th tap changing transformers
$g(x,u) / h(x,u)$: Set of equality constraints/set of inequality constraints
x	: Vector of dependent variables (state variables)
u	: Vector of independent variables (Control Variables)
V_i / V_j	: Voltage magnitude of i -th and j -th bus, respectively
V_{PV} / V_{PQ}	: Voltage vector of generator and load buses, respectively
V_G / V_L	: Voltage at the generator and load buses, respectively

1. INTRODUCTION

The optimal reactive power dispatch (ORPD) is an optimization problem recognized as an important tool in the electrical power engineering area, to manage reactive power in electrical networks. The main objective of ORPD is to assess the optimal operating state of the electrical power grid based on the criteria of economy, service quality and security [1]. The economy and service quality require appropriate voltage control at all buses of the system with tolerable limits to ensure proper reactive power flows and minimal active transmission losses. On the other hand, the security of the power system requires sufficient voltage levels and reactive reserves to prevent voltage stability failures and to maintain the integrity of the power grid in a safe state when critical unforeseen events occur. Power grid security control can be performed by improving the voltage stability margin reflected by the voltage stability index (VSI) or minimizing the total voltage deviation (TVD) from the rated voltage magnitude. The aforementioned goals of ORPD problem can be achieved through the optimal adjustments of all kinds of control variables in the power system given by the voltage magnitude at all buses of generation (continuous control variables), tap setting transformers and reactive power from VAR compensators (discrete control variables). By combining these two types of control variables, the ORPD becomes a mixed integer nonlinear programming (MINLP) optimization problem. The mono-objective resolution of ORPD is stated for minimizing the transmission power active losses (Ploss), to reduce the TVD or to improve the VSI related to load buses, while accomplishing the satisfaction of predefined operational constraints related to the physical system [2].

In literature, many classical optimization methods have been applied to solve the ORPD problem. Linear programming (LP), nonlinear programming (NLP) [3] and Newton method [4] were among the presented techniques in the literature. Unfortunately, these conventional methods present some drawbacks in dealing with non-convex and MINLP optimization problems considering non-differentiable objective functions and constraints, additionally to their premature convergence by trapping in local optima when solving complex optimization problems [5], [6]. Recently, computational intelligence methods have been imposed as an alternative to the classical optimization techniques called meta-heuristics, which are based on mimicking physical or biological phenomena and their main advantage concerns the ability in dealing with combinatorial and non-convex optimization problems. Many of the meta-heuristics have been developed in recent years, and each of them is inspired according to a natural phenomenon. Some of them have been widely employed in solving ORPD problem, such as genetic algorithms (GA) [7], particle swarm optimization (PSO) [8], artificial bee colony (ABC) [9], firefly algorithm (FA) [10], Gravitational search algorithm (GSA) [11], and whale optimization algorithm (WOA) [12].

Therefore, there is no guarantee for a particular meta-heuristic algorithm to reach a perfectly optimal solution and to solve all optimization problems effectively referring to the no free lunch (NFL) theorem [13]. Thus, various research works have been elaborated to enhance the search capability of some meta-heuristics in solving ORPD problem. Radosavljevic and Jevtic [14], studied a combination of gravitational search algorithm and sequential quadratic programming (GSA-SQP) has been implemented as efficient hybrid algorithm to solve ORPD problem in the case of Institute of Electrical and Electronics Engineers (IEEE)-(30buses) test system. This approach tends to avoid a premature convergence of GSA without trapping in local optima. Ghasemi *et al.* [2], studied a modified imperialist competitive algorithm (MICA) was hybridized with invasive weed optimization (IWO) method to improve the optimal solution of the ORPD problem compared to that of the original ICA or IWO method. In the aim to surmount the early convergence problem of the PSO algorithm, Singh *et al.* [15] suggest a more effective alternative method by hybridizing the basic PSO and ant lion optimizer (ALO) algorithm named PSO-ALO with no significantly undermining the fast convergence of (PSO) method. This hybrid approach has proved its success in improving ORPD solution since it finds a better objective functions than most competitive optimization techniques.

Recently, the ABC algorithm inspired by the foraging behavior of honeybees to find and to exploit the nectar of flowers has been extensively applied as an efficient population-based algorithm. Since its development in [16], it has gained great popularity to solve complex optimization problems in various fields of engineering, especially in electrical power system considering economic dispatch (ED) [17], optimal

power flow (OPF) [18] and ORPD [19] due to its simple implementation. Referring to a huge number of works regarding ABC algorithm applications, practical studies have brought into focus that this algorithm is weak in the exploitation of promising solutions and powerful in the exploration of search space [20]-[22]. Conversely, salp swarm algorithm (SSA) inspired from navigating and foraging behaviors of salp swarms living in oceans [22] highlights weak exploration search mechanism and powerful exploitation capability [23], [24]. Therefore, we resorted to improving the exploitation capability of the ABC algorithm by using SSA and developing a hybrid approach based on the ABC algorithm and SSA (ABC-SSA) to improve the optimal solution of the ORPD problem. To the best of our knowledge, this is the first time that this hybrid approach ABC-SSA is suggested and applied to solve ORPD problem. Various test systems are implemented such as IEEE-(30buses) and large system IEEE-(300buses) to confirm the validity of the proposed hybrid ABC-SSA in finding a better solution than using one method at a time and over the proposed hybrid techniques in the literature. The simulation results obtained using ABC-SSA are compared to those of other recently published techniques in literature for the same problem. The presented comparison proves the robustness of this hybrid technique under different cases of study on various scales of power systems. The presented technique promotes its extension to other complex optimization fields.

The main contributions in developing the presented hybrid ABC-SSA technique are as follows:

- The ability of the proposed hybrid ABC-SSA approach to balance between the two searches mechanisms of meta-heuristics (exploitation and exploration) in order to reach better solution of ORPD problem.
- The development of an effective hybrid technique of ABC and SSA algorithms to benefit from advantages of both algorithms.

2. PROBLEM FORMULATION

Generally, the ORPD problem is stated in the following manner:

Minimize $F(x, u)$

$$\text{Subject to: } \begin{cases} g(x, u) = 0 \\ h(x, u) \leq 0 \end{cases} \tag{1}$$

$$x = [V_{L1} \dots V_{NLB}, Q_{G1} \dots Q_{NG}, S_1 \dots S_{NTL}] \tag{2}$$

$$u = [V_{G1} \dots V_{NG}, T_1 \dots T_{NT}, Q_{C1} \dots Q_{CNC}] \tag{3}$$

2.1. Objective functions

2.1.1. Total active transmission losses

The mathematical expression of the active transmission losses in the electrical power network is defined as follows [25]:

$$P_{loss} = \sum_{k=1}^{NTL} G_k \times (V_i^2 + V_j^2 - 2V_i V_j \cos \delta_{ij}) \tag{4}$$

2.1.2. Voltage deviation

The total voltage deviation (TVD) forms an important objective function for electrical network analysis and operation, it represents the sum of voltage magnitude deviations for all load buses concerning their desired values ($V_L^{ref}=1.00 pu$). The minimization of TVD improves voltage profile and enhances the security level of power systems, it is expressed as follows [22]:

$$TVD = \sum_{i=1}^{NLB} |V_{L,i} - V_L^{ref}| \tag{5}$$

2.1.3. Voltage stability index

The improvement of the voltage stability is achieved through the minimization of the voltage stability index (VSI) L_j given by the j -th load node of the electric power grid. In the purpose to enhance voltage stability and to keep the electric power grid so far away from the operating point which provokes the voltage collapse (by improving the stability margin), the maximum of L_j among all load buses is employed as an objective function to minimize for handling the ORPD issue. The voltage stability index L_j of j -th load bus is defined as follows [26]:

$$L_j = \left| 1 - \sum_i F_{ij} \frac{V_i}{V_j} \angle (\theta_{ij} + (\delta_i - \delta_j)) \right| \quad i = 1, 2, \dots, NG \quad j=1, 2, \dots, NLB \tag{6}$$

where: $F_{ij} = |F_{ij}| \angle \theta_{ij}$

$$F_{ij} = -[Y_1]^{-1}[Y_2] \quad (7)$$

δ_i / δ_j = Voltage angle of *bus-i* and *bus-j*, respectively.

Y_1 = Describes the sub-matrix linking the injection current vector and voltage vector of load nodes.

Y_2 = Describes the sub-matrix linking the injection current vector of load nodes and voltage vector of generation nodes.

$$\begin{bmatrix} I_{PQ} \\ I_{PV} \end{bmatrix} = \begin{bmatrix} Y_1 & Y_2 \\ Y_3 & Y_4 \end{bmatrix} \begin{bmatrix} V_{PQ} \\ V_{PV} \end{bmatrix} \quad (8)$$

when the L_j value is closer to zero, the electric power grid is further stable. Represents an equation that maximizes a parameter:

$$L_{max} = \max(L_j) \quad \text{where: } j=1, 2 \dots NLB \quad (9)$$

L_{max} the maximum value of L_j among all load buses.

2.2. Operational constraints

2.2.1. Power flow equality constraints

The power flow for each node of an electrical power grid is characterized by the equality constraints expressed as follows:

$$P_{G,i} - P_{L,i} - \sum_{j=1}^{NB} |V_i| |V_j| |Y_{ij}| \cos(\alpha_{ij} - \delta_i + \delta_j) = 0 \quad (10)$$

$$Q_{G,i} - Q_{L,i} - \sum_{j=1}^{NB} |V_i| |V_j| |Y_{ij}| \sin(\alpha_{ij} - \delta_i + \delta_j) = 0 \quad (11)$$

2.2.2. Operating inequality constraints

The mathematical form of inequality operating constraints is stated in the following manner:

- Constraints description of generator: The output voltage of each generator is characterized by its magnitude, which is limited by upper and lower limits $V_{G,i}^{min}$ and $V_{G,i}^{max}$, respectively.

The reactive power generation is also limited between lower and upper capacity limit $Q_{G,i}^{min}$ and $Q_{G,i}^{max}$, respectively.

$$\begin{cases} V_{G,i}^{min} \leq V_{G,i} \leq V_{G,i}^{max} \\ Q_{G,i}^{min} \leq Q_{G,i} \leq Q_{G,i}^{max} \end{cases} \quad i = 1, 2, 3 \dots NG \quad (12)$$

- The tap setting T_k of transformers is imposed to the restrictions given by lower and upper boundary T_k^{min} and T_k^{max} , respectively. These limits are mathematically given by:

$$T_k^{min} \leq T_k \leq T_k^{max} \quad k = 1, 2, 3 \dots NT \quad (13)$$

- The generated reactive power Q_{ci} from a capacitor bank is confined by two limits Q_{ci}^{min} and Q_{ci}^{max} of lower and upper generation bound, respectively, and expressed as follows:

$$Q_{c,i}^{min} \leq Q_{c,i} \leq Q_{c,i}^{max} \quad i = 1, 2, 3 \dots NC \quad (14)$$

- The power flow rate for each transmission line is confined by its transit capacity limit:

$$S_l \leq S_l^{max} \quad l = 1, 2, 3 \dots N_{NTL} \quad (15)$$

For limiting the dependent variables V_{Li} , Q_G and S_l , we use the technique of penalty factors which prevents the considered dependent variable to go out its limits (by ruling out the solutions providing the limit violations of considered state variable) even if the objective function at these points gives a good solution. For this aim, we use an augmented objective function described by (16):

$$F_{aug} = F(x, u) + \lambda_V \sum_{i=1}^{NLB} \Delta V_{Li} + \lambda_Q \sum_{i=1}^{NG} \Delta Q_{Gi} + \lambda_S \sum_{i=1}^{NTL} \Delta S_{li} \tag{16}$$

where: λ_V , λ_Q and λ_S are the factors of penalty.

$$\Delta V_{Li} = \begin{cases} (V_{Li}^{min} - V_{Li})^2 & \text{if } V_{Li} < V_{Li}^{min} \\ (V_{Li} - V_{Li}^{max})^2 & \text{if } V_{Li} > V_{Li}^{max} \\ 0 & \text{if } V_{Li}^{min} \leq V_{Li} \leq V_{Li}^{max} \end{cases} \tag{17}$$

$$\Delta Q_{Gi} = \begin{cases} (Q_{Gi}^{min} - Q_{Gi})^2 & \text{if } Q_{Gi} < Q_{Gi}^{min} \\ (Q_{Gi} - Q_{Gi}^{max})^2 & \text{if } Q_{Gi} > Q_{Gi}^{max} \\ 0 & \text{if } Q_{Gi}^{min} \leq Q_{Gi} \leq Q_{Gi}^{max} \end{cases} \tag{18}$$

$$\Delta S_{li} = \begin{cases} (S_{li} - S_{li}^{max})^2 & \text{if } S_{li} > S_{li}^{max} \\ 0 & \text{if } S_{li}^{min} \leq S_{li} \leq S_{li}^{max} \end{cases} \tag{19}$$

3. ARTIFICIAL BEE COLONY ALGORITHM

The artificial bee colony algorithm as an interesting meta-heuristic optimization technique has proven its efficiency for solving various numerical optimization problems in the engineering area [18]. It has been inspired by the honeybees activities to collect the nectar of food sources and to share roles during the foraging process. The ABC algorithm considers the food source position as a proposed solution in the search space, while the nectar quantity of food source corresponds to the fitness value of the potential solution. The hive population is divided in two groups of bees: employed and unemployed bees, where each group contains the half population of the hive. The employed bees are sent to search for food sources, while the unemployed bees, called onlooker bees, are waiting in the hive to receive information about food sources discovered by employed bees. Once onlooker bees have received information about food sources, they try to select the best ones among them (with high quantity of nectar) to further explore the vicinity of the best food source positions (exploiting the best solutions). When a food source is exhausted, its employed bee changes the role to become a scout bee, which tries to find a new food source in other location. Three phases are performed to accomplish one cycle after the initialization phase of the population. A predefined number of cycles can be selected as stopping criteria of the ABC algorithm (16-18-20).

3.1. Initializing a population of solutions

Initially, random positions of SN food sources are generated in the hive environment using the following equation:

$$x_{ij} = x_{j,min} + R_n \cdot (x_{j,max} - x_{j,min}) \tag{20}$$

where $x_{j,max}$, $x_{j,min}$ are, the upper and lower limits of j -th decision variable in the D -dimensional search space respectively with $i \in [1, 2, .. SN]$, $j \in [1, 2, .. D]$, and R_n is a randomly generated number in the interval $[0, 1]$.

3.2. Exploiting food sources by employed bees

After the initialization of food source locations, all employed bees are sent to discover the food sources in the neighborhood of the previously memorized food source positions. Each employed bee tries to find a food source around an old one in its memory. This behavior is modeled mathematically using the following equation:

$$v_{ij} = x_{ij} + \Phi_{ij}(x_{ij} - x_{Kj}) \tag{21}$$

where v_{ij} and x_{ij} indicate the new and the old j -th variable related to the i -th position of the food source, respectively, with $i \in \{1, 2, \dots, SN\}$ and $j \in \{1, 2, \dots, D\}$. x_{Kj} is the j -th variable of the k -th position of food source chosen randomly. Φ_{ij} is a randomly generated real number between -1 and 1. If v_{ij} in (21) violates its predefined limits, it is fixed to its violated limit. A greedy selection process is carried out to select between x_i and v_i .

3.3. Exploiting food sources by onlooker bees

Once the employed bees have accomplished their investigation phase, they will communicate with onlooker bees about all information of food sources, particularly, the positions and nectar quantities. Each onlooker bee must choose one food source based on the probability evaluation p_i corresponding to this food source. Using the probability of roulette wheel, p_i can be evaluated by the following expression:

$$P_i = \frac{fitness_i}{\sum_{j=1}^{SN} fitness_j} \quad (22)$$

$fitness_i$: Fitness value of solution i

The vector from the population of solutions $x_i = [x_{i1}, x_{i2}, \dots, x_{iD}]$ is evaluated by calculating its corresponding objective function $f(x_i) = f_i$ and the fitness function $fitness_i$ is given by:

$$fitness_i = \begin{cases} \frac{1}{1+f(x_i)} & \text{if } f(x_i) \geq 0 \\ 1 + abs(f(x_i)) & \text{if } f(x_i) < 0 \end{cases} \quad (23)$$

The onlooker bee searches the neighborhood of selected food source position x_i in order to produce a new candidate solution by changing one parameter in the vector x_i using (21). The new generated solution v_i is evaluated referring to (23). Then the greedy selection takes part again in this case, to retain the best solution and rejecting that of poor quality.

3.4. Exchanging role of the employed bee to a scout bee

After the full exploitation of a food source, its corresponding employed bee becomes a scout bee and it will change the location to look for a new food source in the search space. This stage is reached when a proposed solution x_i has not improved after a predetermined number of trials named "limit" and based a Trial Counter (TC) corresponds to each potential solution. TC_i of i -th solution is incremented by 1 if no improvement of this solution, else TC_i is reset to zero. Thus, the new food source is generated randomly using (20). The control parameter "limit" can be used as a key factor to avoid the ABC algorithm to be trapped in local minima during the search process. The ABC algorithm steps are illustrated as below:

- Step 1: Set the ABC algorithm parameters SN , $limit$, D and $maxCycle$.
- Step 2: Creating an initial random population (food sources) using (20).
- Step 3: Evaluating each food source by determining the fitness value using (23) and reset TC to zero for each one.
- Step 4: Start $Cycle = 1$.
- Step 5: Start the phase of employed bees.
for $i = 1$ to SN
 discover a new food source position v_i depending on the old one x_i by applying (21) where ($k \neq i$),
 evaluate each new food source using (23), apply the greedy selection to choose between x_i and v_i ,
 increment TC_i by 1 if no improvement of the i -th food source, else reset TC_i to zero.
end for
- Step 6: Calculating the probability p_i for each bee using (22).
- Step 7: Start the phase of onlooker bees.
for $i = 1$ to SN
 Generate a random value Rn
 if $Rn < p_i$
 discover a new food source position v_i depending on the old one x_i by applying (21) where ($k \neq i$),
 evaluate each new food source using (23), apply the greedy selection to choose between x_i and v_i ,
 increment TC_i by 1 if no improvement of the i -th food source, else reset TC_i to zero.
 end if
end for
- Step 8: Start the scout bee phase
for $i = 1$ to SN
 if $TC_i > limit$

Create a new food source emplacement x_i using (21)
 end if
 end for

Step 9: Store the global best food source obtained until now.

Step10: Verifying if $Cycle > maxCycle$, if yes exit by stopping the algorithm execution, otherwise do $Cycle = Cycle + 1$ and go to step 5.

4. SALP SWARM ALGORITHM

The imitation of the SSA is from the attitude of salps belonging to salpidae species, living in oceans and possessing a transparent body in the form of a barrel like a jelly-fish. The salps move with pumped water through their body to propel themselves forward Figure 1 (a). It is believed that the salps move so that they organize a salp chain in oceans and seas searching the best sources of food as shown in Figure 1 (b).

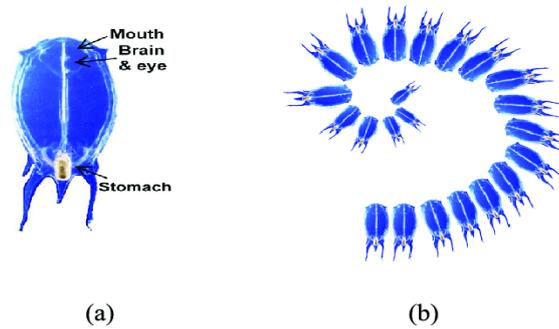


Figure 1. (a) Individual salp, (b) Swarm of salps (salps chain)

To model the salp chain behavior in the mathematical aspect, the salp swarm is partitioned in two sub-populations of leader and followers. By leading the salp chain, the leader tries to govern the displacement of the followers. Each follower of the salp chain tracks the path mapped by one leader. In a similar manner as other categories of optimization techniques founded on the swarm attitude, the salp position is expressed in the search space with D -dimensions, where D reflects the number of control variables relating to the optimization problem. Consequently, Np positions of salps are memorized in a matrix X with two-dimensions. By assuming that the target of the swarm is a food source designated by F_j , the salp chain attempt to reach it during the search process [27], [28]. To deal with the update of the leader position, the (24) can be suggested:

$$X_j^1 = \begin{cases} F_j + c_1 ((ub_j - lb_j)c_2 + lb_j) & c_3 \geq 0.5 \\ F_j - c_1 ((ub_j - lb_j)c_2 + lb_j) & c_3 \leq 0.5 \end{cases} \quad (24)$$

where:

- X_j^1 = Leader position in j -th dimension relating to the first salp.
- F_j = The position occupied by the best food source in the dimension j at each iteration.
- ub_j and lb_j = Are upper and lower limits in the dimension j of D -dimensional space, respectively.
- c_2, c_3 : = Two randomly generated numbers in the interval between 0 and 1.

In (24) exposes the update of the leader position by referring to the food source position. The factor c_1 represents the key factor in the harmonization between two important mechanisms of meta-heuristics exploration and exploitation during the research for good solutions specified in the following equation:

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \quad (25)$$

By considering:

l : Current iteration, L : Maximum number of iterations.

For updating the follower salp position, this task is accomplished according to the suggested (26).

$$X_j^i = \frac{1}{2}(X_j^i + X_j^{i-1}) \quad (26)$$

where the index i must be greater or equal to 2 and X_j^i depicts the i -th follower salp position in j -th dimension. By employing (24) and (26), the simulation process of salps in regrouped chains can be mimicked. Referring to mathematical inspiration, the principal steps of the SSA algorithm are shown below:

- Step 1: Set SSA parameters like as D, N_p, ub, lb, L and initialize $l=1$.
- Step 2: Create a random population of solutions by initializing the positions of salps.
- Step 3: Evaluate the population for the objective function and find the best global solution F (food source).
- Step 4: Dividing the salp population in two sub-populations of leaders and flowers.
- Step 5: Updating iteration number $l=l+1$.
- Step 6: Calculate the constant c_1 based (25), while c_2 and c_3 are generated randomly.
- Step 7: Update leader and follower positions using (24) and (26), respectively.
- Step 8: Amending salp positions referring to lower and upper limits of variables.
- Step 9: Update the food source F .
- Step 10: Verifying if $l < L$ go to step5, else extract the best global solution F and exit from the algorithm execution.

5. HYBRID ABC-SSA APPROACH

All meta-heuristic optimization techniques try to balance between their two important mechanisms: exploration and exploitation. The exploration is related to the search capacity of the algorithm in finding encouraging new solutions, while exploitation is associated with the capability of the algorithm to discover an optimum near the best solution. Referring to (21), which describes the search mechanism of the ABC algorithm, the new generated position v_i (new solution) moves away from (or near) the old position (solution x_i) depending on the selection probability p_i in (23). This behavior tends to improve the exploration capability. The main disadvantage affecting the ABC algorithm is the update of position x_i based only one search (21) by changing one parameter (one variable related to this solution). In addition, when the greedy selection is applied between x_i and v_i , the bad solution is ignored without giving it more chance of exploitation. Therefore, the ABC method is good at exploration but poor at exploitation. The SSA uses the (24) in order to update the position of the leader referring to the best solution F_j (food source), which enhance the exploitation capability to find promising solutions in the vicinity of the optimal solution found so far F_j . The hybrid ABC-SSA approach tries to improve the exploitation proficiency of the ABC algorithm and that by introducing the bad solutions Sn_i which have not been improved (extracted using greedy selection between x_i and v_i in the employed and onlooker bee phases) in SSA. In such manner, these solutions are more exploited using the local search process by SSA in order to improve the solution quality of the optimization problem. The best global solution achieved by the ABC algorithm is stored as food source F_j in the SSA. To support all these stages, the hybrid ABC-SSA steps are described below:

- Step 1: Firstly, set the ABC algorithm parameters (N, D and $maxCycle$) and accomplishing ABC method steps (from step 2 until step 3).
- Step 2: Start Cycle =1.
- Step 3: Begin by step 4 until step 8 in the ABC steps (employed bees phase and onlooker bees phase).
- Step 4: Extracting the population of solutions Sn_i which has not been improved during ABC method and memorizing the best global solution achieved so far by ABC algorithm as best food source F_ABC .
- Step 5: Starting with SSA and evaluate the population of solutions Sn_i using (24).
- Step 6: Complete the same steps mentioned in the SSA method (from step 5 until step 9), and memorize the best food source as F_SSA .
- Step 7: Compare between F_ABC and F_SSA and extract the best food source among them.
- Step 8: Check if $Cycle < maxCycle$, if Yes do $Cycle = Cycle + 1$ and then go to step 3, else extract the best food source and exit from the program.

6. SIMULATION RESULTS AND DISCUSSIONS

To investigate the enhancement of the proposed ABC-SSA approach in solving the ORPD problem, two standard test systems are considered which are IEEE-(30buses), and IEEE-(300buses). The mono-objective optimization issue is stated by minimizing the total active power losses (Ploss), voltage stability index (L-index) or total voltage deviation (TVD). Table 1 presents the characteristics of the test systems. The population size (N), the maximum number of iterations (Max_iteration) and penalty factors λ_v ,

λ_Q in (16) for each test power system are given in Table 2. The optimal solution achieved by the developed algorithm (ABC-SSA) is selected for the best solution over thirty runs independently executed.

Table 1. Description of test power systems

Description	IEEE-(30buses)	IEEE-(300buses)
Number of control variables	19	190
Number of Generators	6	69
Number of Taps	4	107
Number of Q-shunt	9	14
Equality constraints	60	530
Inequality constraints	125	706
Discrete variables	6	107
Ploss (MW)	5.81	408.316
TVD (pu)	0.5821	5.4286

Table 2. Control parameter settings of ABC, SSA, and ABC-SSA algorithms for test power systems

Algorithm	ABC, SSA, and ABC-SSA			
	λ_V	λ_Q	N	Max_iteration or maxCycle
IEEE-30 bus	0	0	80	150
IEEE-300 bus	10^2	10-6	500	500

6.1. IEEE-(30 buses) system

The first test system implemented in the ORPD problem is that of IEEE-(30buses). it contains 19 control variables including 6 for generator voltage magnitude outputs located in buses 1; 2; 5; 8; 11 and 13, 4 for tap setting transformers connected between buses (6–9, 6–10, 4–12 and 28–27), and 9 for reactive power output from shunt capacitors in buses 10; 12; 15; 17; 20; 21; 23; 24 and 29. The data base for this network is mentioned in [29], the total real power demand is 2.834 (pu) at 100 MVA Base. The limit of the control variables is shown in [29].

6.1.1. Active power losses minimization for IEEE-(30buses) system

In this case, Ploss is selected as an objective function to minimize and the best control variables resulting from ABC-SSA computing code running are shown in Table 3, the results established are compared with those of other available methods in literature as PSO [29], comprehensive learning particle swarm optimization (CLPSO) [29], WOA [12] and improved gravitational search algorithm by conditional selection strategies IGSA_CS [30], as well as the implementation of the developed approaches in this paper ABC-SSA. The minimum obtained Ploss from the ABC-SSA algorithm is 4.5578 MW and it is less by 0.1152 MW (2.53%) than SSA, which gives 4.6730 MW. The convergence curves for ABC, SSA, and ABC-SSA methods are illustrated in Figure 2, which demonstrates that the new hybrid approach does not have any stagnations for the global best solution evolution as it does for SSA and ABC methods. This characteristic shows better performances of ABC-SSA in tackling the premature convergence.

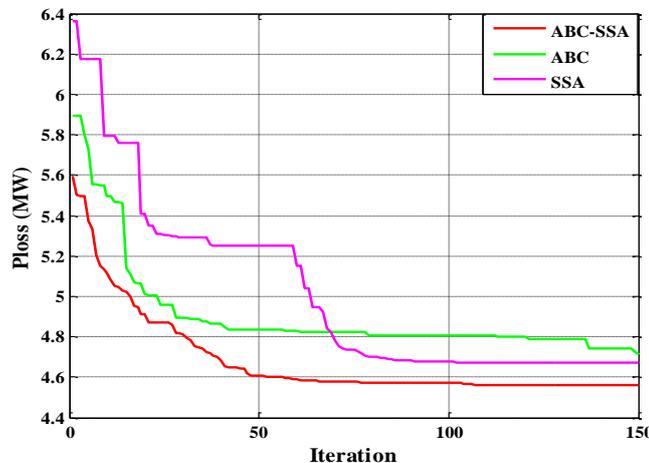


Figure 2. Convergence curves for Ploss minimization, IEEE-(30buses)

Table 3. Simulation results using ABC-SSA and other optimization techniques for Ploss minimization-IEEE-(30buses)

control variables	ABC-SSA	ABC	SSA	PSO [29]	CLPSO [29]	WOA [12]	IGSA_CS [30]
Generator voltage							
V ₁	1.1000	1.0578	1.1000	1.1000	1.1000	1.1000	1.081281
V ₂	1.0942	1.0565	1.0945	1.1000	1.1000	1.0963	1.072177
V ₅	1.0738	1.0236	1.0749	1.0867	1.0795	1.0789	1.050142
V ₈	1.0762	1.0176	1.0768	1.1000	1.1000	1.0774	1.050234
V ₁₁	1.1000	1.0426	1.0707	1.1000	1.1000	1.0955	1.100000
V ₁₃	1.1000	1.0686	1.0814	1.1000	1.1000	1.0929	1.068826
Transformer tap ratio							
T ₆₋₉	1.0684	1.0380	1.0147	0.9587	0.9154	0.9936	1.080
T ₆₋₁₀	0.9000	1.0289	1.0036	1.0543	0.9000	0.9867	0.902
T ₄₋₁₂	0.9998	1.0755	1.0593	1.0024	0.9000	1.0214	0.990
T ₂₈₋₂₇	0.9760	1.0396	1.0040	0.9755	0.9397	0.9867	0.976
Capacitor banks							
Q _{C-10}	5.0000	2.7614	4.3881	4.2803	4.9265	3.1695	0.00
Q _{C-12}	5.0000	2.0468	4.3416	5.00	5.0000	2.0477	0.00
Q _{C-15}	5.0000	0.9966	2.6818	3.0265	5.0000	4.2956	0.0380
Q _{C-17}	5.0000	2.7687	1.6046	4.0365	5.0000	2.6782	0.0490
Q _{C-20}	5.0000	4.5165	1.4919	2.6697	5.0000	4.8116	0.0395
Q _{C-21}	5.0000	3.3702	2.7040	3.8894	5.0000	4.8163	0.0500
Q _{C-23}	3.8635	3.5046	2.7881	0.0000	5.0000	3.5739	0.0275
Q _{C-24}	5.0000	2.4227	4.3374	3.5879	5.0000	4.1953	0.0500
Q _{C-29}	3.1172	3.8632	2.4971	2.8415	5.0000	2.0009	0.0240
P _{loss} MW	4.5578	4.7157	4.6730	4.6282	4.5615	4.5943	4.76601
TVD (pu)	1.8117	0.4789	1.0592	1.0883	0.4773	-	-
L-index (pu)	0.1169	0.1452	0.1277	0.1423	0.1230	-	-

6.1.2. TVD minimization for IEEE-(30buses) system

In this case, TVD is the objective function to be minimized for the same IEEE-(30buses) test network. Table 4 presents the results deduced from the simulation stage, which includes the best optimum (TVD) with the proposed ABC-SSA algorithm and the two implemented approaches ABC and SSA. A comparison is made with other optimization methods provided in literature, like, hybrid firefly algorithm (HFA) [31], PSO [29], and CLPSO [29]. Therefore, as shown in Table 4, there is a TVD improvement of 5.15% than the best result obtained by HFA [31] that gives 0.098 pu. Figure 3 illustrates the convergence characteristics of each method confirming the fastest convergence rate of ABC-SSA in reaching the best global optimum.

Table 4. Comparison of simulation results for IEEE-(30buses) with TVD minimization objective

Algorithms	ABC-SSA	ABC	SSA	HFA [31]	PSO [29]	CLPSO [29]
Ploss MW	6.0744	6.0945	5.7451	5.75	4.7075	4.6969
TVD (pu)	0.0932	0.1097	0.1053	0.098	0.2577	0.2450
L-index(pu)	0.1369	0.1374	0.1367	-	0.1273	0.1247

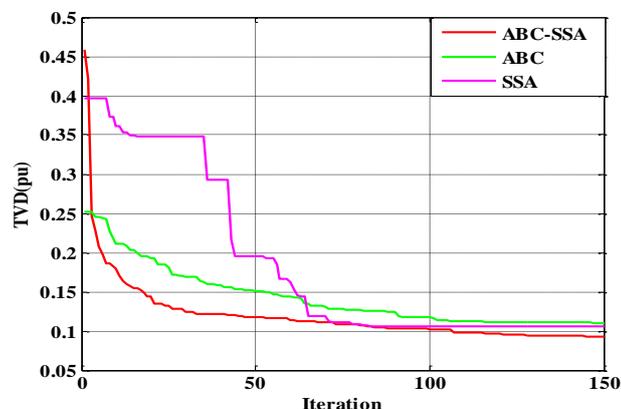


Figure 3. Convergence curves for TVD minimization of IEEE-(30buses) power system

6.1.3. VSI improvement for IEEE-(30buses) system

In order to enhance the margin stability of IEEE-(30buses) system, the VSI given by max (L-index) is minimized by applying the ABC-SSA. The simulation results are listed in Table 5 and compared with those of ABC, SSA, GSA [11], and opposition-based gravitational search algorithm OGSA [32]. The optimal value of VSI using ABC-SSA is better than that of the OGSA method in the literature signalling a remarkable reduction (important improvement) equal to 9.82%.

Table 5. Comparison of simulation results for IEEE-(30buses) test power system with improvement of VSI

Algorithms	ABC-SSA	ABC	SSA	OGSA [32]	GSA [11]
P _{loss} MW	4.7173	5.4268	4.6581	5.9198	4.975298
TVD (pu)	2.1341	1.8551	2.0349	1.9887	0.215793
L-index(pu)	0.1120	0.1138	0.1139	0.1230	0.136844

6.2. IEEE-(300 buses) system

To test and prove the applicability of the SSA algorithm in more practical, complicated and large test systems, IEEE-(300buses) is proposed which consists of large-scale dimensions of control variables containing 190 control variables where their types are demonstrated in Table 6. The total load data are 235.258pu and 77.8797 pu for active and reactive power, respectively [33], [34]. The control variables restrictions are indicated in [35].

For the first case of this test system application, the total active losses (P_{loss}) are minimized using ABC-SSA algorithm. Table 1 reports the best results found by the ABC-SSA method and other optimization approaches as ABC, SSA, specialized genetic algorithm (SGA), and ALO methods. The optimal solution of the objective function (P_{loss}) was obtained with an improvement equal to 3.05 % when it is compared to that of the SGA method. The search mechanism for the optimal solution was clarified by a convergence curve in Figure 4 related to the ABC-SSA, ABC, SSA. It is clearly remarked that the convergence profile using ABC-SSA is the promising one. Figure 5 exhibits the voltage profile when the optimal solution is achieved for the present test system, noting that the voltage magnitude at all buses is in its permissible range without any violations beyond the permissible limits.

Table 6. Comparison of simulation results for IEEE 300-Bus with ploss minimization

Results	ABC-SSA	ABC	SSA	SGA [35]	ALO [36]
P _{loss} MW	343.4272	364.347	353.886	357.10	384.922
TVD (pu)	20.3408	1.5292	18.2697	15.744	-
L-index (pu)	0.9003	0.8879	0.8340	-	0.3663

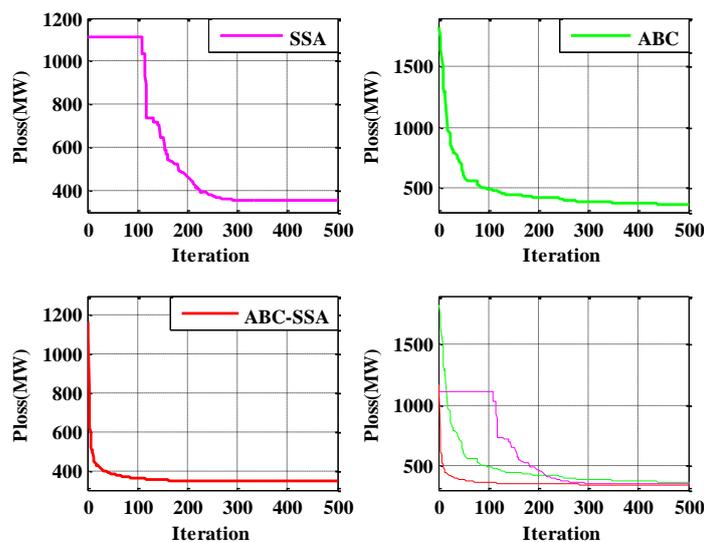


Figure 4. Convergence curves for Ploss minimization IEEE-(300buses)

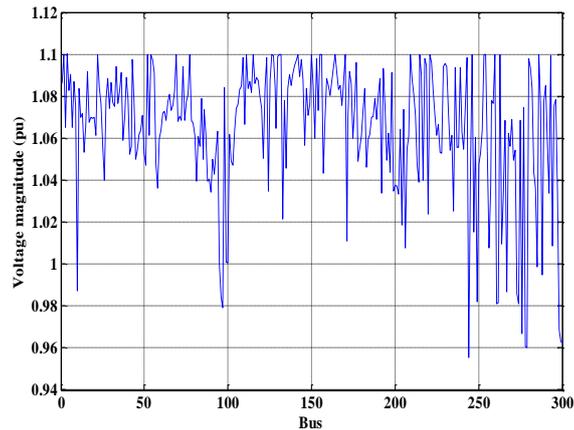


Figure 5. Bus voltage profile for IEEE-(300buses) power system

7. CONCLUSION

In this paper, a new hybrid bio-inspired optimization approach combining artificial bee colony (ABC) and Salp Swarm (SSA) algorithms named (ABC-SSA) was developed and successfully employed for solving different problems of optimal reactive power dispatch (ORPD) with several types of complexities. The presented approach was examined and evaluated regarding different objective functions. The effectiveness and robustness of the novel ABC-SSA are investigated using two standard test systems IEEE. A comparison report of ABC-SSA with the original ABC and SSA algorithms is made based on convergence curves. A smooth convergence curve is devoted to ABC-SSA approach when it is compared to that of basic ABC and SSA algorithms, which proves the capacity of the proposed approach to escape from the stagnation in local minima and to converge in faster manner towards the global optimal solution. Another comparison survey of the ABC-SSA with different optimization techniques in the same literature is provided. The results of the simulation report prove that the ABC-SSA offers better performances than other comparison methods, indicating the robustness and the superiority of the ABC-SSA, which shows a remarkable exploitation capability by using the best solution (food source of SSA) at each iteration to achieve promising solutions. Thus, the ABC-SSA algorithm can be recommended as a promising optimization algorithm in solving other more complex optimization problems for engineering area, particularly in electrical power systems.

REFERENCES

- [1] P. K. Pany, S. Ghoshal, and V. Mukherjee, "Collective animal behavior algorithm for optimal reactive power dispatch problems," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017, pp. 1327-1332, doi: 10.1109/ICPCSI.2017.8391926.
- [2] M. Ghasemi, S. Ghavidel, M. M. Ghanbarian, and A. Habibi, "A new hybrid algorithm for optimal reactive power dispatch problem with discrete and continuous control variables," *Applied soft computing*, vol. 22, pp. 126-140, 2014, doi: 10.1016/j.asoc.2014.05.006.
- [3] A. Chebbo and M. Irving, "Combined active and reactive despatch. i. problem formulation and solution algorithm," *IEE Proceedings-Generation, Transmission and Distribution*, vol. 142, pp. 393-400, 1995, doi: 10.1049/ip-gtd:19951976.
- [4] M. Bjelogrić, M. S. Calović, P. Ristanović, and B. S. Babić, "Application of Newton's optimal power flow in voltage/reactive power control," *IEEE Transactions on Power Systems*, vol. 5, pp. 1447-1454, 1990, doi: 10.1109/59.99399.
- [5] R. Salimin, I. Musirin, Z. Hamid, A. Harun, S. Suliman, H. Suyono, and R. Hasanah, "Multi cases optimal reactive power dispatch using evolutionary programming," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 17, pp. 662-670, 2019, doi: 10.11591/ijeecs.v17.i2.pp662-670.
- [6] K. Lenin, B. R. Reddy, and M. Suryakalavathi, "Reduction of Real Power Loss and Safeguarding of Voltage Constancy by Artificial Immune System Algorithm," *Bulletin of Electrical Engineering and Informatics*, vol. 4, pp. 88-95, 2015, doi: 10.11591/eei.v4i2.436.
- [7] P. A. Jeyanthi and D. Devaraj, "Optimal Reactive Power Dispatch for Voltage Stability Enhancement Using Real Coded Genetic Algorithm," *International Journal of Computer and Electrical Engineering*, vol. 2, p. 734, 2010.
- [8] M. Abido, "Optimal power flow using particle swarm optimization," *International Journal of Electrical Power & Energy Systems*, vol. 24, pp. 563-571, 2002, doi: 10.7763/IJCEE.2010.V2.220.
- [9] K. Ayan and U. Kılıç, "Artificial bee colony algorithm solution for optimal reactive power flow," *Applied soft computing*, vol. 12, pp. 1477-1482, 2012, doi: 10.1016/j.asoc.2012.01.006.
- [10] G. Kannan, D. P. Subramanian, and R. U. Shankar, "Reactive power optimization using firefly algorithm," in *Power Electronics and Renewable Energy Systems*, ed: Springer, 2015, pp. 83-90.

- [11] S. Duman, Y. Sonmez, U. Guvenc, and N. Yorukeren, "Optimal reactive power dispatch using a gravitational search algorithm," *IET generation, transmission & distribution*, vol. 6, pp. 563-576, 2012, doi: 10.1049/iet-gtd.2011.0681.
- [12] K. ben oualid Medani, S. Sayah, and A. Bekrar, "Whale optimization algorithm based optimal reactive power dispatch: A case study of the Algerian power system," *Electric Power Systems Research*, vol. 163, pp. 696-705, 2018, doi: 10.1016/j.epsr.2017.09.001.
- [13] T. T. Nguyen and D. N. Vo, "Improved social spider optimization algorithm for optimal reactive power dispatch problem with different objectives," *Neural Computing and Applications*, vol. 32, no. 6, pp. 1-32, 2019, doi: 10.1007/s00521-019-04073-4.
- [14] J. Radosavljevic and M. Jevtic, "Solution of optimal reactive power dispatch by a hybrid GSA-SQP algorithm," *Elektronika ir Elektrotehnika*, vol. 22, pp. 3-6, 2016, doi: 10.5755/J01.EIE.22.3.15306.
- [15] R. P. Singh, V. Mukherjee, and S. Ghoshal, "Optimal reactive power dispatch by particle swarm optimization with an aging leader and challengers," *Applied soft computing*, vol. 29, pp. 298-309, 2015, doi: 10.1016/j.asoc.2015.01.006.
- [16] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Technical report-tr06, Erciyes University*, 2005.
- [17] M. Basu, "Artificial bee colony optimization for multi-area economic dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 49, pp. 181-187, 2013, doi: 10.1016/j.ijepes.2013.01.004.
- [18] M. R. Adaryani and A. Karami, "Artificial bee colony algorithm for solving multi-objective optimal power flow problem," *International Journal of Electrical Power & Energy Systems*, vol. 53, pp. 219-230, 2013, doi: 10.1016/j.ijepes.2013.04.021.
- [19] S. Mouassa and T. Bouktir, "Artificial bee colony algorithm for discrete optimal reactive power dispatch," in *2015 International Conference on Industrial Engineering and Systems Management (IESM)*, 2015, pp. 654-662, doi: 10.1109/IESM.2015.7380228.
- [20] C. Zhang, J. Zheng, and Y. Zhou, "Two modified artificial bee colony algorithms inspired by grenade explosion method," *Neurocomputing*, vol. 151, pp. 1198-1207, 2015, doi: 10.1016/j.neucom.2014.04.082.
- [21] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied mathematics and computation*, vol. 217, pp. 3166-3173, 2010, doi: 10.1016/j.amc.2010.08.049.
- [22] A. Salhi, D. Naimi, and T. Bouktir, "Optimal power flow resolution using artificial bee colony algorithm based grenade explosion method," *Journal of Electrical Systems*, vol. 12, pp. 734-756, 2016.
- [23] N. Singh, F. Chiclana, and J.-P. Magnot, "A new fusion of salp swarm with sine cosine for optimization of non-linear functions," *Engineering with Computers*, vol. 36, pp. 185-212, 2020, doi: 10.1007/s00366-018-00696-8.
- [24] R. A. Ibrahim, A. A. Ewees, D. Oliva, M. A. Elaziz, and S. Lu, "Improved salp swarm algorithm based on particle swarm optimization for feature selection," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 3155-3169, 2019.
- [25] P. L. Reddy and G. Yesuratnam, "A modified bacterial foraging algorithm based optimal reactive power dispatch," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 13, p. 361367, 2019, doi: 10.11591/ijeecs.v13.i1.pp361-367.
- [26] M. Basu, "Quasi-oppositional differential evolution for optimal reactive power dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 78, pp. 29-40, 2016, doi: 10.1016/j.ijepes.2015.11.067.
- [27] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163-191, 2017, doi: 10.1016/j.ijepes.2015.11.067.
- [28] H. Faris *et al.*, "An efficient binary Salp Swarm Algorithm with crossover scheme for feature selection problems," *Knowledge-Based Systems*, vol. 154, pp. 43-67, 2018, doi: 10.1016/j.knosys.2018.05.009.
- [29] K. Mahadevan and P. Kannan, "Comprehensive learning particle swarm optimization for reactive power dispatch," *Applied soft computing*, vol. 10, pp. 641-652, 2010, doi: 10.1016/j.asoc.2009.08.038.
- [30] G. Chen, L. Liu, Z. Zhang, and S. Huang, "Optimal reactive power dispatch by improved GSA-based algorithm with the novel strategies to handle constraints," *Applied Soft Computing*, vol. 50, pp. 58-70, 2017, doi: 10.1016/j.asoc.2016.11.008.
- [31] A. Rajan and T. Malakar, "Optimal reactive power dispatch using hybrid Nelder-Mead simplex based firefly algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 66, pp. 9-24, 2015, doi: 10.1016/j.ijepes.2014.10.041.
- [32] B. Shaw, V. Mukherjee, and S. Ghoshal, "Solution of reactive power dispatch of power systems by an opposition-based gravitational search algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 55, pp. 29-40, 2014, doi: 10.1016/j.ijepes.2013.08.010.
- [33] A. P. Mazzini, E. N. Asada, and G. G. Lage, "Solving control-constrained reactive power dispatch with discrete variables," in *2015 18th International Conference on Intelligent System Application to Power Systems (ISAP)*, 2015, pp. 1-6, doi: 10.1109/ISAP.2015.7325566.
- [34] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on power systems*, vol. 26, pp. 12-19, 2010, doi: 10.1109/TPWRS.2010.2051168.
- [35] W. Villa-Acevedo, J. López-Lezama, and J. Valencia-Velásquez, "A Novel Constraint Handling Approach for the Optimal Reactive Power Dispatch Problem," *Energies*, vol. 11, no. 9, p. 2352, 2018, doi: 10.3390/en11092352.
- [36] S. Mouassa, T. Bouktir, and A. Salhi, "Ant lion optimizer for solving optimal reactive power dispatch problem in power systems," *Engineering science and technology, an international journal*, vol. 20, pp. 885-895, 2017, doi: 10.1016/j.jestch.2017.03.006.