[1]Nguyen Tung Linh

[2]Truong Viet Anh

[3]Pham Vu Long

A Novel Combine Crow Search Algorithm and Particle Swarm Optimization for the Problem of Reconfiguration Distribution Networks Considering Distributed Generation and Electric Vehicles

Journal of Electrical Systems

Abstract: - In modern times, with the increasing prevalence of renewable energy sources and electric vehicle systems integrated into distribution networks, the operation to ensure power quality, optimization, and reliability of distribution networks has become more critical. The reconfiguration problem of the distribution network, aiming to reduce losses, optimize renewable energy sources, and regulate the charging/discharging process of electric vehicles, is a highly necessary task. The Particle Swarm Optimization (PSO) algorithm excels in exploitation but is less effective in exploration. In contrast, the Crow Search Algorithm (CSA) is simple and more random, will improve the speed and accuracy of optimization problems. Thus, this study proposes using a combination of the PSO algorithm and the CSA algorithm applied to the distribution network reconfiguration problem, considering renewable energy sources and electric vehicles. The research results, validated on the IEEE 33-bus system with various scenarios, show that the proposed method is accurate and reliable.

Keywords: Reconfiguration distribution network, Power losse, Particle Swarm Optimization (PSO), Crow Search Algorithm (CSA), Renewable energy, Electric vehicle.

I. INTRODUCTION

The increasing demand for energy, coupled with the gradual depletion of fossil fuel reserves and severe climate change, has posed significant challenges to modern societies. In recent years, the development of renewable energy sources and the use of electric vehicles (EVs) have significantly increased due to their ability to reduce pollution and lower operating costs. Renewable energy sources (RES), such as photovoltaic (PV) systems, have played a vital role in reducing environmental pollution in recent years by mitigating the greenhouse effect [1]. As an advanced and widely used method of power generation, photovoltaic power production aligns with sustainable development strategies and the concept of safe electricity generation. With the rise of distributed generation (DG), PV can now operate at a smaller scale known as distributed renewable energy sources (RES). This form of PV is designed to be closer to electricity loads, using a decentralized investment model to reduce transmission losses [2].

However, PV systems are characterized by intermittent and instability. Solar radiation, cloud cover, the orientation of PV panels, dust diffusion, and other factors can significantly disrupt PV system operations [3]. Furthermore, high penetration of PV can lead to issues such as voltage rise, reverse current flow, and increased energy losses [4].

Distribution network reconfiguration (DNR) is an effective technique to enhance the integration of distributed generation (DG) and electric vehicles (EVs) into distribution networks, ensuring the target objectives and

[1] [1*]Faculty of Control and Automation, Electric Power University, Ha Noi, Viet Nam

linhnt@epu.edu.vn https://orcid.org/0000-0003-2645-4599

[2]Faculty of Electrical Engineering University of Technology and Education Ho Chi Minh, Viet Nam

anhtv@hcmute.edu.vn, https://orcid.org/0000-0002-5151-5771

[3]Institute of Energy, 6 Ton That Tung Hanoi, Vietnam Postgraduate Electric Power University, longpv@ievn.com.vn,

https://orcid.org/0009-0000-7373-8063

***Correspondening Author:** Nguyen Tung Linh

*Faculty of Control and Automation, Electric Power University, Ha Noi, Viet Nam

linhnt@epu.edu.vn https://orcid.org/0000-0003-2645-4599

technical conditions are met. In distribution networks, reconfiguration is performed through switching control of breakers (circuit breakers, isolating switches, etc.)[5]. During reconfiguration, the network's feeders are updated to achieve specific goals such as reducing power losses, minimizing operating costs, or enhancing reliability based on the switching status of switching devices. This process must also ensure operational constraints are not violated, maintaining power quality standards while avoiding isolating parts of the distribution network.

Integrating DG and EVs into the grid brings significant improvements such as reducing power losses, peak shaving, and reducing voltage drops. However, due to the uncertainty and randomness of DG and EVs, changing the distribution network configuration is necessary [6]. Given the evident advantages of this process with the involvement of the aforementioned factors, its importance is emphasized. Therefore, many studies have focused on analyzing DNR with the combination of RES, EVs, and energy storage (ES) systems.

Several studies have been conducted to reconfigure the power network in the presence of static Distributed Generators (DG). The Gravitational Search Algorithm (GSA) has been applied to solve the Distribution Network Reconfiguration (DNR) problem, aiming to improve reliability and reduce power losses [7]. GSA has also been used to enhance transient stability indices and reduce losses in DNR. Moreover, a combination of the Particle Swarm Optimization (PSO) algorithm and the Shuffled Frog Leaping Algorithm (SFLA) has been developed to improve voltage stability indices and reduce operating costs when solving DNR with the presence of DG units [8-9]. Two population-based evolutionary optimization methods have been proposed to find the optimal placement of DG units, helping reduce losses and operating costs.

Experimental studies show that most power outages occur in the distribution network across the entire electrical system due to the instability of new energy sources and electric vehicles [10]. Severe power outages pose significant problems in many countries, affecting the scale of the distribution network [11]. To improve reliability and reduce power losses, one approach is to leverage the characteristics of open-loop operation and the closed-loop structure in the distribution network [11-12]. Distribution Network Reconfiguration (DNR) maintains substation voltage levels in the distribution network and enhances operational efficiency by changing the on/off status of tie switches [13]. Thus, maintaining normal operations and reducing power losses through DNR is essential, especially with the penetration of new energy sources and electric vehicles [14].

A new meta-heuristic algorithm, sin-cosine, has been used to reconfigure the distribution network with simultaneous DG allocation [15]. An effective meta-heuristic method, Tabu Search (TS), has also been modified to quickly find optimal solutions [16].

In this study, the Particle Swarm Optimization (PSO) algorithm excels in exploitation but is less effective in exploration. In contrast, the Crow Search Algorithm (CSA) is simple and more random. This study introduces a new crow swarm optimization algorithm (CSO) that combines the strengths of PSO and CSA. The new algorithm allows individuals to explore unknown regions by following the guidance of a randomly chosen individual. The simulation results are conducted on the IEEE 33-bus test system with various scenarios, such as considering load variation, the output power of RES sources, and based on the load profile of EV loads. In addition to the general introduction, Section 2 presents the problem model, Section 3 discusses the proposed method, Section 4 includes the simulation on the test grid, and Section 5 provides an evaluation and conclusion.

II. PROBLEM FORMULATION

A. Optimal objective function

1) The function power loss:

Reconfiguration of the distribution network is a challenging combinatorial optimization problem. When considering multiple objective functions, the network's various performance metrics include minimizing load deviation, voltage deviation, and system power losses [10].

$$F_1 = \min \sum_{i=1}^{n} k_i R_i \frac{P_i^2+Q_i^2}{U_i^2} \quad (1)$$

where n represents the aggregate number of branches; k_i denotes the switch position (with $k_i = 0$ indicating open and 1 indicating closed); R_i is the overall resistance of the ith branch; and P_i, Q_i, and U_i correspond to the terminal active power, reactive power, and node voltage at the termination of branch i, respectively [11], [12].

2) The function voltage deviation

$$F_2 = \min \sum_{j=1}^{n} (\frac{U_j - U_{js}}{U_{js}})^2 \qquad (2)$$

where n is the total number of nodes; U_j is the actual voltage of node j; U_{js} is the rated voltage of node j.

3) The variance of the function load

$$F_3 = \min \sum_{i=1}^{m} (\frac{S_i}{S_{imax}})^2 \qquad (3)$$

where m is the total number of closed branches; S_i and S_{imax} respectively represent the actual value and the maximum value of the complex power on branch i.

4) Objective function normalization

The method of random weight allocation has been employed to normalize the objective function.

$$\omega_i = \frac{rand_i}{\sum_{i=1}^{n} rand_i} \qquad (4)$$

$$F = \min(\omega_1 \frac{F1}{f_1} + \omega_2 \frac{F2}{f_2} + \omega_3 \frac{F3}{f_3}) \qquad (5)$$

where ω_i represents the random weight coefficient assigned to the i^{th} objective function, rand generates random numbers within the interval [0,1], and F_i is the minimum value achieved by the i^{th} objective function in each iteration.

B. The DG mathematical model

1) Solar photovoltaic modeling

The following formula can be used to calculate solar power output power [13]:

$$P_{PV} = \eta P_{rate} \frac{A}{A_s} [1 + \alpha p(T - T_{STC})] \qquad (6)$$

where η denotes the power factor, P_{rate} refers to the rated power, A indicates the actual light intensity, A_s is the light intensity in standard test conditions, αp represents the power temperature coefficient, T is the present surface temperature of the photovoltaic cell, and T_{STC} is the temperature of the photovoltaic cell during standard test conditions.

2) Wind turbine modeling

The main factor influencing wind energy's output power is wind speed, which is best explained as follows:

$$P_t(v) = \begin{cases} 0, & 0 \le v \le V_{ci} \\ av^3 - bP_{r,} & v_{ci} \le v \le v_r \\ P_{r,} & v_r \le v \le v_{co} \\ 0, & v_{co} \le v \end{cases} \qquad (7)$$

where P_r represents the rated power, and v_{ci}, v_r, and v_{co} denote the minimum, rated, and maximum wind speeds for power generation, respectively, while v stands for the current wind speed. The output from distributed generation (DG) is often simplified and handled as a 'negative load,' being treated as a series of continuous variables. When the active power and power factor of a DG are known, it can be considered as a P, Q node.

$$\begin{cases} P = -P_s \\ Q = -Q_s \end{cases} \qquad (8)$$

where P_s and Q_s indicate, respectively, active power and reactive power of DG.

C. Technical binding conditions

1) Power balance restrictions [14]

$$\begin{cases} P_G - P_i - U_i \sum_{j=1}^{n} U_i\left(G_{i-j}\cos\theta_{i-j} + B_{i-j}\sin\theta_{i-j}\right) = 0 \\ Q_G - Q_i - Q_i \sum_{j=1}^{n} U_i\left(G_{i-j}\cos\theta_{i-j} + B_{i-j}\sin\theta_{i-j}\right) = 0 \end{cases} \quad (9)$$

where P_G and Q_G refer to the active and reactive power injected at the DG node, respectively, while P_i and Q_i indicate the active and reactive power at the load node.

2) Operational constraints

$$\begin{cases} U_{min} \le U_i \le U_{max} \\ I_{min} \le I_i \le I_{max} \\ S_i \le S_{imax} \\ P_G \le P_{Gmax} \end{cases} \quad (10)$$

where U_{min} and U_{max} represent the voltage limits at node i, setting the minimum and maximum thresholds, respectively. I_{min} and I_{max} define the lowest and highest current limits for branch i. S_i refers to the complex power on branch i, while P_{Gmax} indicates the highest output power that branch i can accommodate in the power distribution system.

III. PROPOSED METHOD FOR PROBLEM

The study introduces a new Crow Swarm Optimization (CSO) algorithm combining the strengths of Particle Swarm Optimization (PSO) and the Crow Search Algorithm (CSA). The CSO balances the strong exploitation of PSO with the simple, random exploration of CSA, allowing individuals to explore unknown areas guided by random peers. Testing the CSO on various benchmark functions shows that it enhances optimization efficiency, global search ability, and robustness, outperforming PSO and CSA, particularly in high-dimensional, complex problems.

A. Over view PSO

The Particle Swarm Optimization (PSO) algorithm was developed by Eberhart and Kennedy in 1995. Since then, it has been widely applied in control systems and optimization calculations [18-20]. The trajectory of each individual in the search space is adjusted by changing its velocity, based on its flight experience and that of other individuals in the search space. The position vector and velocity vector of an individual i in a multi-dimensional space are:

$$x_i = (x_{11}; x_{12}; x_{13}; \dots.; x_{1n};);\ V_i = (v_{11}; v_{12}; \dots.; v_{1n};) \quad (11)$$

At each iteration, the velocity of a particle is determined by both personal experience and the experience of the entire group:

$$V_i^{t+1} = \omega.v_i^t + c_1.r_1(pb_i^t - x_i^t) + c_2.r_2(gb_i^t - x_i^t) \quad (12)$$

$$i = 1,2, \dots. n;\ v_{min} \le v_i^{t+1} \le v_{max}$$

$$x_i^{k+1} = X_{id}^k + V_{id}^{k+1} \quad (13)$$

$$x_{min} \le x_i^{t+1} \le x_{max}$$

where:

V_i^{t+1} is the velocity vector of particle i.

X_i^{t+1} is the position of particle i.

pb_i^t is the best previous index of each individual.

gb_i^t is the best index among all individuals in the population.

k is the number of iterations the individuals have moved.

c_1 and c_2 are acceleration constants.

r_1 and r_2 are two random numbers uniformly distributed within the range [0, 1].

v_{min} and v_{max} are the lower and upper limit of particle update velocity, respectively; in this paper, $v_{min} = -v_{max}$. x_{min} and xmax are the minimum and maximum positions of particles, respectively.

Equation 12 is the first term represents the previous velocity, giving the particle momentum to continue wandering through the search space. The second component is considered the cognitive element, representing the artificial intelligence of the particles. This part directs the particles toward their best positions. The third component is the social element, which represents the combined effect of particles in finding the global optimal solution. The social element guides particles toward the global optimum. Initially, particles are generated at random positions, and each particle is assigned a random velocity. The fitness of the particles is estimated through the objective function. At each point, the velocity of each particle is calculated, and its position is updated for the next estimation using (13). Over time, if the particles find a better position than their previous one, this new position is saved in memory.

B. Over view CSA

The crow search algorithm (CSA) is a new metaheuristic optimization method that simulates the intelligent behavior of crow flocks. Introduced by Askarzadeh in 2016, this algorithm showed promise in its initial results, demonstrating the potential to tackle a variety of complex engineering optimization problems. The Crow Search Algorithm (CSA) is a swarm intelligence optimization strategy that mimics the behavior of crows tracking each other [23]. This paper outlines the fundamental principles of the algorithm, examines the primary parameters influencing its performance, and analyzes its search mechanism. Improvement strategies are summarized, and examples are provided to demonstrate the algorithm's application in the 0-1 knapsack problem, image processing, scheduling issues, feature selection, and parameter optimization. Finally, considering the current research and application of the crow search algorithm, future research and development directions are discussed.

The principles of CSA include the following: (1) crows live in flocks, (2) each crow remembers where its hiding places are and can steal food from other crows, and (3) a crow may protect its food by flying randomly if it realizes it's being followed, with a certain probability [24]. The position of each crow represents a solution x_i^t. In each iteration, crow i tracks a randomly chosen crow j. If crow j is unaware of being followed (i.e., rj $\geq$AP), crow i approaches crow j's hiding place pb_j^t. However, if crow j senses it is being tracked, it deceives crow i by flying to a random location in the search space to safeguard its hiding place. The mathematical expression is:

$$\begin{cases} x_i^{t+1} = x_i^{t+1} + r_3 . fl . \left(pb_j^t - x_i^t\right), & r_j \geq AP \\ \text{a random position} & \text{else} \end{cases} \quad (14)$$

where, r_j and r_3 are random numbers uniformly distributed between 0 and 1, fl represents the length of a crow's flight, AP is crow j perceptual probability, and pb_j^t indicates where the current crow j stores its food, which corresponds to the historical best solution crow j has found.

The Crow Search Algorithm (CSA) is a population-based optimization method that is relatively simple, with only two adjustable parameters [25]: flight length fl and perceived probability AP. This simplicity makes it attractive for various engineering applications. In CSA, perceived probability parameters directly control the algorithm's diversity. Compared to the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Harmony Search Algorithm (HS), CSA has fewer parameters to adjust and is easier to implement. Additionally, individual crows in CSA can reach entirely random positions, giving them a stronger capacity to explore unknown regions.

However, CSA lacks criteria for choosing destinations, as selections are made randomly between crows, and the flight length is constant. These characteristics lead to a weaker ability to exploit current information compared to PSO, resulting in lower search precision, a higher likelihood of falling into local optima, and premature convergence, particularly in multi-dimensional optimization problems.

C. Proposed combine PSO and CSA for problem

To leverage the strengths of both PSO in exploitation and CSA in exploration, this study introduces a new Crow Swarm Optimization (CSO) algorithm. In CSO, particle movement is guided by the best position discovered by the individual particle and the best position found by the entire swarm, similar to PSO. At the same time, each particle monitors others, meaning its movement may be influenced by the best position discovered by the swarm and another particle's best solution. In CSA, this involves tracking where another crow hides its food. The equation for updating the flight velocity of crows in CSO is:

$$\begin{cases} v_i^{t+1} = \omega . v_i^t + c_3 . r_3\left(pb_j^t - x_j^t\right) + c_2 . r_2\left(gb_j^t - x_j^t\right) & r_j \geq AP \\ v_i^{t+1} = \omega . v_i^t + c_1 . r_1\left(pb_j^t - x_j^t\right) + c_2 . r_2\left(gb_j^t - x_j^t\right) & \text{else} \end{cases} \quad (15)$$

where c_3 represents the degree of the influence of individual j on individual i and r_3 is a random number within [0, 1]. The updating velocity use (14) is also limited by vmin and vmax.

When $r_j \geq AP$ individual i chooses to follow individual j, and its velocity is influenced by its inertia velocity, the global optimal solution, and the current optimal solution of individual j. Otherwise, the velocity of individual i is determined by its inertia velocity, the global optimal solution, and its own local optimal solution. Once the velocity of individual i is calculated, its position in the next iteration is obtained using (13).

Fig 1 and Fig 2 present the flowchart CSO and pseudocode of the proposed algorithm CSO application reconfiguration distribution network consider DG and Evs. The primary difference between the CSO and the other two algorithms lies in how particle speed and position are updated. PSO focuses on optimization efficiency and aims to approach the current best solution during each iteration, leading to strong exploitation of existing information. On the other hand, CSA provides more freedom to the algorithm to maintain solution diversity, which enhances its ability to explore new regions. The proposed CSO combines the strengths of both PSO and CSA, achieving a better balance between randomness and efficiency, or in other words, between exploration and exploitation.

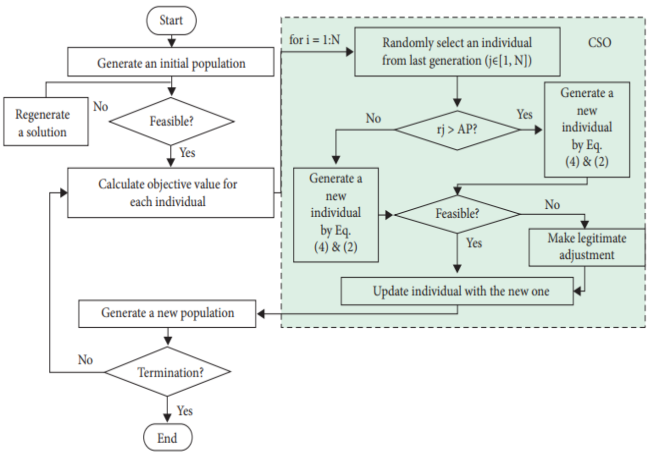

Fig. 1. Flowchart of the proposed CSO for problems

```
# Initialize the algorithm by reading system data and defining parameters
Read system data: flock size, flight length,
Pareto optimal set size, consciousness probability
Set max iterations and max objectives
Generate an initial flock of random individuals
Initialize Pareto set as empty
# Begin the iteration loop
while iteration_count < max_iterations:
    # Initialize the flight count for each crow
    flight_count = 1
    Initialize the position and memory for each crow randomly
    # Perform load flow check
    Run load flow
    Check for any functional constraint violations
    # Evaluate each flight objective
    while flight_count <= flock_size:
        Compute objectives (e.g., obj1, obj2)
        Evaluate flight fitness (e.g., fit1, fit2)
        Increment flight count
    # If all flights have been assessed
    if K == flock_size:
        # Determine non-dominated solutions in the current flock
        current_flock = get_current_flock()
        non_dominated_set = determine_non_dominated_solutions(current_flock)
        pareto_set = update_pareto_set(non_dominated_set)
    # Check for feasibility and generate a new solution
    if size(current_flock) < max_size:
        # Combine current and Pareto flocks
        combined_flock = combine_flocks(current_flock, pareto_set)
        # Update the position and memory of each crow
        Update_position_and_memory(combined_flock)
    else:
        pareto_set = cluster_and_reduce_pareto_set(pareto_set)
    # Increment the iteration count
    iteration_count += 1
    # For each individual in the population, apply the CSO steps
    for i in range(1, N):
        # Randomly select a crow from the previous generation
        j = select_random_crow(i, N)
        # Apply the CSO logic
        if r_j > AP:
            # Generate a new individual using Equations (14) and (13)
            new_individual = generate_new_individual(Equation(14), Equation(13))
        else:
            # Adjust to ensure feasibility
            new_individual = adjust_for_feasibility(Equation(14), Equation(13))
        # Check the feasibility of the new individual
        if feasible(new_individual):
            Update_individual_with_new(new_individual)
# Use fuzzy min-max to find the best compromise solution
final_solution = find_best_solution(pareto_set)
Stop and output the final solution
```

Fig. 2. Pseudocode of the proposed algorithm CSO application RDN consider DG and Evs.

Individual Movement: The movement of individuals significantly impacts the performance of swarm intelligence algorithms. Fig. 2 illustrates how individual positions are updated in (a) PSO, (b) CSA, and (c) CSO. In PSO, movement is consistent and determined by a particle's inertia, its best current position, and the best position identified by the whole swarm. Both CSA and CSO offer alternatives with varying probabilities to better sustain solution diversity. However, CSA is less efficient than CSO due to differences in the way solutions are updated.

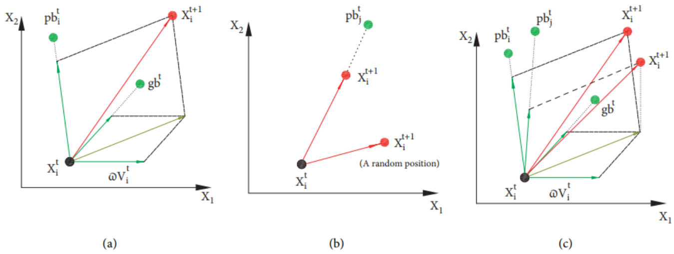

Fig. 3. Schematics of how an individual updates its position in (a) PSO, (b) CSA, and (c) CSO

CSO maintains PSO's optimization efficiency while providing opportunities to explore broader regions. In CSO, the higher the AP probability parameter value, the more directionally oriented the movement becomes. When AP equals 1, CSO reverts to standard PSO; when AP equals 0, individuals always randomly choose a historical optimal solution to follow, with less emphasis on leveraging other known information. Notably, when AP is set to 0, CSO does not transform into CSA but retains the rich diversity characteristic of CSA.

IV. EXPERIMENTS AND RESULTS

The proposed CSO algorithm was applied for evaluation on the 33-bus IEEE distribution system sample shown in Fig. 4, [13]. The initial network parameters include a total load of 3715 + j2300 kVA, consisting of 33 nodes, 37 branches, and 5 switching devices, with further details available in the study by Goswami and [14]. The system includes two renewable energy sources: a wind turbine at node 15 with a capacity of 700 kW and another renewable source at node 30 with a capacity of 500 kW. Additionally, the charging stations at nodes 10 and 23 each serve 30 electric vehicles (EVs), with each EV having a battery capacity of 35.9 kWh.

In this study, the simulation results are evaluated in two different scenarios. Scenario 1: The PV and WT power sources are assumed to have stable output at 60% of their rated capacity. Scenario 2: The output power of PV and WT depends on weather characteristics such as temperature, radiation, and wind speed. The EV load also varies based on the usage needs of EVs, so this uncertainty is simulated using a power graph.

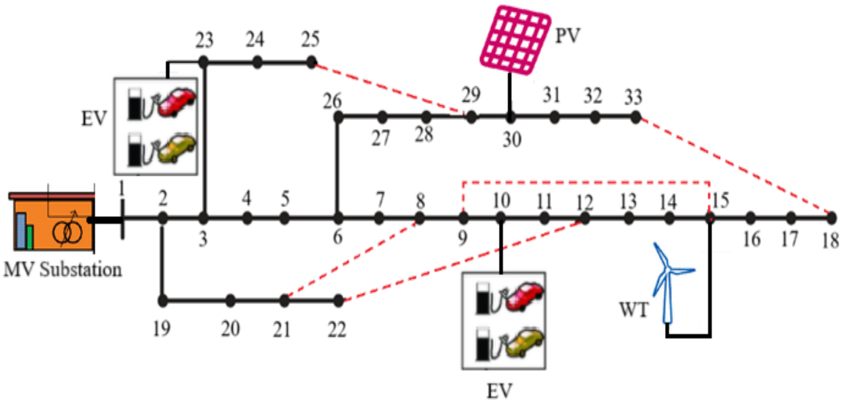

Fig. 4. IEEE sample distribution network - 33 bus.

A. The result for scenario 1

The simulation results for scenario 1 conducted in four different cases including:

Case 1. Before reconfiguration without DG.

Case 2. After reconfiguration without DG.

Case 3. Before reconfiguration with DG.

Case 4. After reconfiguration with DG.

The results of the study are presented in Table 1 and Fig 5.

TABLE I. RESULTS OF THE CASES

Case	**Open switches**	**P_{losse}(kW)**	**Voltage deviation/p.u**	**F_{min} (use Eq.5)**
1	8–21, 9–15; 12–22, 18–33, 25–29	203.15	0.104	198.83
2	7–8, 14–15; 9–10, 32–33, 25–29	137.92	0.051	132.77
3	8–21, 9–15,12–22, 18–33, 25–29	109.82	0.053	97.83
4	7–8, 14–15; 9–10, 31–32, 25–29	82.73	0.031	73.23

Table I reveals that network losses prior to reconfiguration without distributed generation (DG) were 203.15 kW, dropping to 137.92 kW post-reconfiguration, resulting in a reduction of approximately 32.109%. This led to notable improvements in load balancing and voltage deviation, enhancing the operational stability of the distribution network and demonstrating the effectiveness of the reconfiguration method presented in this paper.

With DG, network losses decreased from 109.82 kW before reconfiguration to 82.73 kW afterward, reflecting a reduction of around 32.74%.

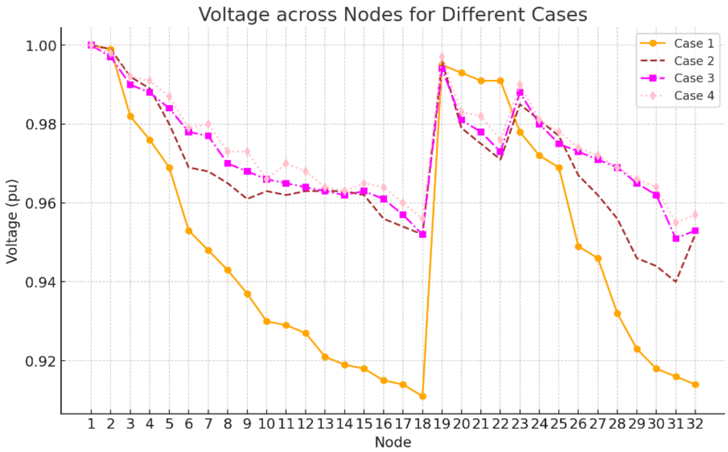

Fig. 5. Graph voltage of IEEE 33- distribution network in the cases different

Fig 5 indicates that integrating DG successfully lowered network losses, improved voltage margins, and minimized voltage deviations. To evaluate the proposed method's effectiveness, it was tested on the IEEE 33-node grid model incorporating DG and EV. The results were compared with the methods described by Chen [13] and Liu [14], demonstrating comparable switch closure/opening results to those achieved in previous studies. The simulation outcomes are detailed in Table II.

TABLE II. THE OUTCOMES OF THE METHOD

Method	**Opened switchs**	**Plosse (kW)**	**Node Voltage (pu)**
Before reconfiguration	8–21; 9–15; 12–22; 18–33; 25–29	202.68	0.9131
Proposed method (CSO)	**7–8; 9–10; 14–15; 25–29; 32–33.**	**138.89**	**0.9389**
Liu [14]	7–8; 9–10; 14–15; 25–29; 32–33.	139.57	0.9378
D. Zhang [15]	7–8; 9–10; 14–15; 25–29; 32–33.	139.55	0.9379

B. The result for scenario 2

In practice, DG sources such as wind turbines (WT) and solar power (PV) have output power that depends on various uncertain factors, leading to variations in power output based on these uncertainties, such as wind distribution and wind speed. Moreover, the loads from electric vehicles (EVs) also have random characteristics.

Therefore, in this section, the simulation results are obtained when DG and EVs have randomly varying power outputs.

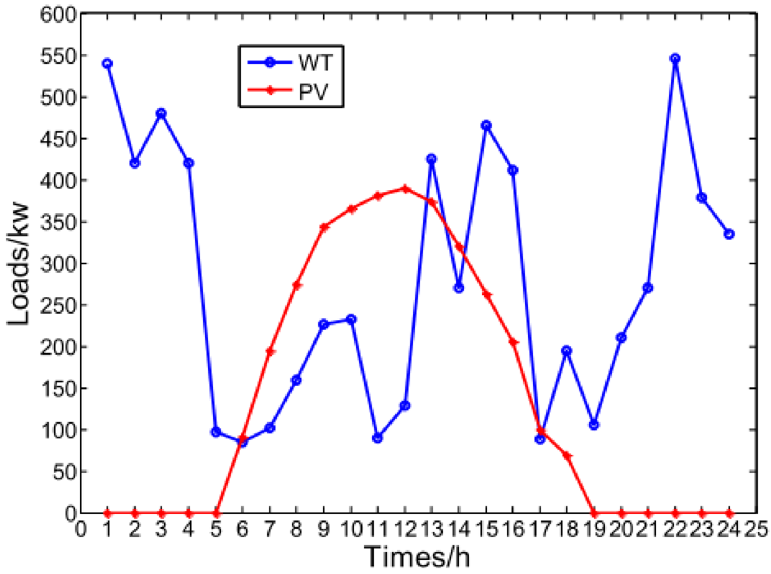

Fig. 6. Forecasting the output power of DG (WT-PV)

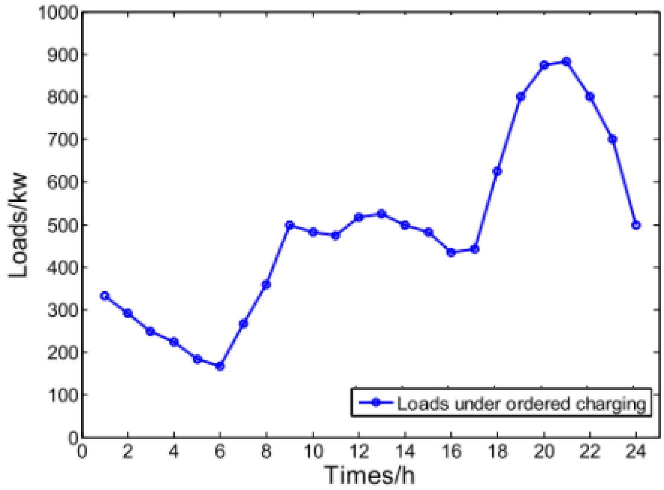

Fig. 7. Forecasting the power consumption of EVs

In the study by Luo [13], the DGs are connected as shown in Figure 4, and the forecasted output of the PV and WT is plotted in Fig. 6. Additionally, the charging power of EVs over 24 hours is projected as shown in Fig. 7.

Considering the load type, proportion, and output at each node [14], the daily load variation is shown in Fig. 8.

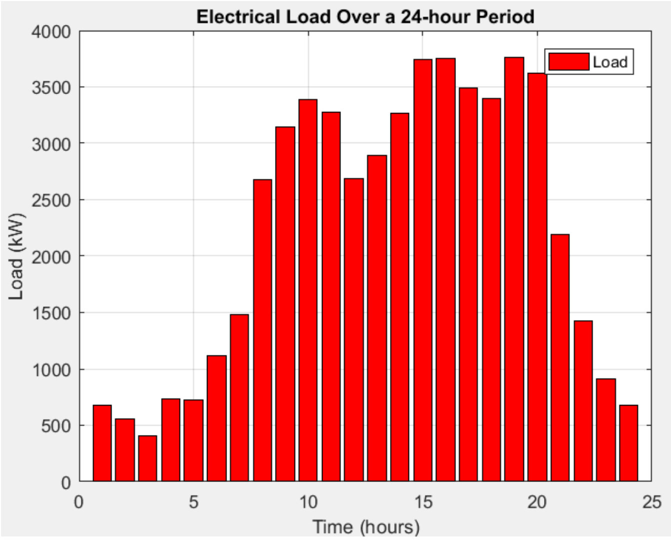

Fig. 8. Load curve of the day for the IEEE 33-node.

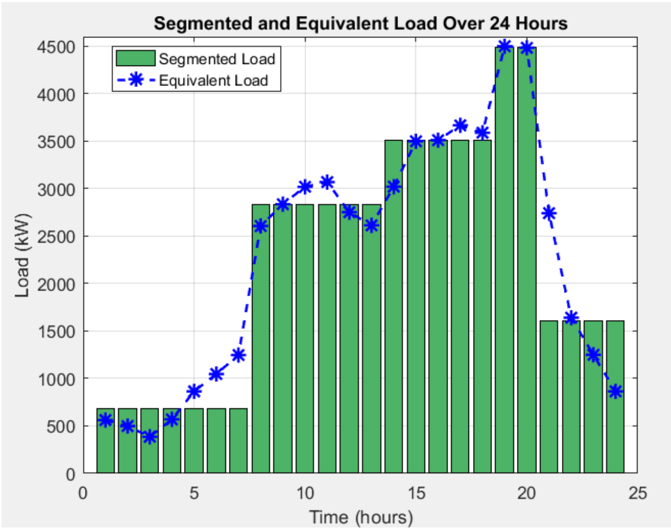

Fig. 9. Hourly/daily dynamic load segmentation graph.

A corresponding daily load curve is established, and the information entropy period division method is applied to segment the load curve as shown in Fig.9. The segmentation in Fig. 9 is used for dynamic reconstruction, resulting in the following outcomes.

Based on Tables III when renewable energy sources (WT, PV) change and load capacity fluctuates over 24 hours, the grid reconfiguration differs at each point in time. However, the structure between the configurations remains similar, with only one or two switching devices varying. Power losses fluctuate at each moment, showing that the dynamic reconfiguration approach has optimized power losses over the 24-hour period. The maximum and minimum voltage values at the nodes all fall within the allowable limits.

TABLE III. RECONFIGURATION RESULTS ACCORDING TO THE LOAD CURVE

Hours	**Open switches**	**P_{losse} (kW)**	**U_{max}/U_{min} (pu)**
1:00 –7:00	7–8, 12–13, 8–9,31–32, 25–29	25.855	0.998/0.9321
8:00 –13:00	7–8, 13–14, 9–10, 32–33, 25–29	380.675	0.998/0.9452
14:00 –18:00	7–8, 13–14, 9–10, 32–33,	518.661	0.998/0.9411

	25–29		
19:00 –20:00	7–8, 9–15, 10–11, 32–33, 25–29	330.522	0.998/0.9386
21:00 –24:00	7–8, 9–15, 9–10, 31–32, 25–2	71.121	0.998/0.9402

TABLE IV. RESULTS NUMBER OF SWITCH OPERATION RECONFIGURATION

Dynamic Reconfiguration	**Number of switch operation**	**P_{losse} (kW)**
No segment load	45	1308.236
Segment load	7	1324.672

Additionally, Table IV illustrates that the method of grid reconfiguration based on load segmentation substantially decreases the frequency of switching operations and the operational costs associated with the switching devices, while also reducing the downtime caused by the state changes of these devices.

V. CONCLUSION

In this study, the authors examined the problem of reconfiguring the distribution grid with a multi-objective function, which considers power loss, reliability, and economic factors in terms of the number of switching operations of the devices. The proposed combination of the CSA and PSO algorithms has facilitated fast and accurate convergence, avoiding local optima. Additionally, in the simulation results section, the authors conducted simulations with uncertain distributed power sources such as WT and PV, dependent on weather conditions, and with the participation of EVs in the load. The research findings indicate that changing the configuration when the output power of the DGs changes and EVs charge/discharge is necessary for optimal operation of the distribution system.

REFERENCES

[1] Shaheen, A. Elsayed, A. Ginidi, R. El-Sehiemy and E. Elattar, “Improved Heap-Based Optimizer for DG Allocation in Reconfigured Radial Feeder Distribution Systems”. IEEE Syst. J. vol 99, pp 1–10. 2022, DOI: 10.1109/JSYST.2021.3136778

[2] H.A. Taha, and M.H. Alham, “Multi-Objective Optimization for Optimal Allocation and Coordination of Wind and Solar DGs, BESSs and Capacitors in Presence of Demand Response” IEEE Access vol 10, pp. 16225–16241, 2022. DOI: 10.1109/ACCESS.2022.3149135

[3] D.R. Prabha, T. Jayabarathi, R. Umamageswari and S. Saranya, “Optimal location and sizing of distributed generation unit using intelligent water drop algorithm”. Sustain. Energy Technol Assess. vol 11, pp. 106–113. 2015, DOI: 10.1016/j.seta.2015.07.003

[4] R. Sanjay, T. Jayabarathi, T. Raghunathan, V. Ramesh and N. Mithulananthan, “Optimal a location of distributed generation using hybrid grey Wolf optimizer”. IEEE Access, vol 5, pp. 14807–14818, 2017. DOI: 10.1109/ACCESS.2017.2726586

[5] Uniyal and S. Sarangi, “Optimal network reconfiguration and DG allocation using ada tive modified whale optimization algorithm considering probabilistic load flow” Electr. Power Syst. Res, vol 192, pp - e106909. 2021.

[6] K. Qian, C. Zhou, M. Allan, and Y. Yuan, “Modeling of load demand due to EV battery charging in distribution systems” IEEE Transactions on Power Systems, vol 26 (issue 2), pp 802–810, 2010, https://doi.org/10.1109/TPWRS.2010.2057456

[7] T.H.B. Huy, T.V. Thanh, V.N. Dieu and H.T.T. Thao “An improved metaheuristic method for simultaneous network reconfiguration and distributed generation allocation” Alexandria Eng. J. vol 61, pp. 8069–8088. 2022, https://doi.org/10.1016/j.aej.2022.01.056

[8] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, “GSA: A gravitational search algorithm”. Information Sciences, vol 179 (no.13), pp. 2232–2248. 2009, https://doi.org/10.1016/j.ins.2009.03.004

[9] Z. Xu, J. Pan, and S. Fan, "A distribution network reconfiguration method with distributed generation based on improved firefly algorithm". Power System Protection and Control, vol 46 (issue14), pp. 26–32, 2018, https://doi.org/10.1080/21642583.2020.1833781

[10] X. Liu, Z. Gao, S. Tian, Z. Wang, and D. Wu, "Optimized reconfiguration of distribution network with distributed power". Optimized Reconfiguration of Distribution Network with Distributed Power, vol 14 (issue 7), pp 815–860, 2019

[11] Shaheen, A. Elsayed, A. Ginidi, R. El-Sehiemy andE. Elattar "Reconfiguration of electr cal distribution network-based DG and capacitors allocations using artificial ecosystem optimizer: Practical case study". Alexandria Eng. J, vol 61, pp. 6105–6118. 2022, DOI: 10.1016/j.aej.2021.11.035

[12] N.T. Linh; P.V. Long, "A Novel Solution Method for the Distribution Network Reconfiguration Problem based on an Objective Function and considering the Cost of Electricity Transmission", Engineering, Technology & Applied Science Research (ETASR). vol.13, No 6 pp: 12366-12372, 2023, doi.org/10.48084/etasr.6568

[13] J.P. Liu, T.X. Zhang, J. Zhu and T.N. Ma, "Allocation optimization of electric vehicle charging station (EVCS) considering with charging satisfaction and distributed renewables integration". Energy vol. 164, pp. 560–574. 2018, DOI: 10.1016/j.energy.2018.09.028

[14] L. Luo, "Reconfiguration of distribution network with distributed generations and electric vehicles". Journal of Electric Power, vol 34 (issue2), pp. 109–116, 2019,

[15] D. Zhang, T. Zhang, X. Xu, Y. Zhou, and X. Zhang, "Optimal reconfiguration of the active distribution network with distributed generation and electric vehicle". Journal of Engineering, vol 13, pp. 1453–1456. 2018, https://ieeexplore.ieee.org/document/8311214.

[16] Chen, Y. Wang, B. Liu, Y. An, and X. Huang, X. "A fast network reconfiguration method avoiding infeasible solutions for distribution system". Transactions of China Electrotechnical Society, vol 30, pp. 34–43, 2015, doi:10.19595/j.cnki.1000-6753.tces.2015.07.005

[17] J. Del Ser, E. Osaba, and D. Molina , "Bio-inspired computation: where we stand and what's next," Swarm and Evolutionary Computation, vol. 48, pp. 220–250, 2019. https://doi.org/10.1016/j.swevo.2019.04.008

[18] N.T. Linh., "A Novel Combination of Genetic Algorithm, Particle Swarm Optimization and Training and Learning-Based Optimization for Distribution Network Reconfiguration in Case of Faults", Engineering, Technology & Applied Science Research (ETASR), vol 14, Issue 1, pp 12959-12965, 2024, https://doi.org/10.48084/etasr.6718

[19] H. Yang, Y. Peng, and N. Xiong, "A static method for distribution network dynamic reconfiguration". Power System Protection and Control, vol. 37 (issue 8), pp. 53–57, 2009,

[20] J. Zhang and X. Yuan, "A novel genetic algorithm based on all spanning trees of undirected graph for distribution network reconfiguration" Advanced Materials Research, vol 960, pp. 943–951, 2014, https://doi.org/10.4028/www.scientific.net/AMR.960-961.943

[21] H. Liu, S. Shi, P. Yang, J. Yang, "An improved genetic algorithm ap-proach on mechanism kinematic structure enumeration with intelligent manufacturing", J. Intell. Robot. Syst. vol. 89, pp. 343– 350, 2017.

[22] Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm" Computers & Structures, vol. 169, pp. 1–12, 2016, https://doi.org/10.1016/j.compstruc.2016.03.001.

[23] Qu and Y. Fu, "Crow search algorithm based on neighborhood search of non-inferior solution set," IEEE Access, vol. 7, pp. 52871–52895, 2019, DOI: 10.1109/ACCESS.2019.2911629

[24] F. Mohammadi and H. Abdi, "A modified crow search algorithm (MCSA) for solving economic load dispatch problem," Applied Soft Computing, vol. 71 (issue 3), pp. 51–65, 2018, DOI: 10.1016/j.asoc.2018.06.040.

[25] S. Makhdoomi and A. Askarzadeh, "Optimizing operation of a photovoltaic/diesel generator hybrid energy system with pumped hydro storage by a modified crow search algorithm," Journal of Energy Storage, vol. 27, Article ID 101040, 2020, https://doi.org/10.1016/j.est.2019.101040.