Optimization of Substation Siting and Connection Topology in Offshore Wind Farm Based on Modified Firefly Algorithm

Zhicong Huang, *Member, IEEE*, Canjun Yuan, Hanchen Ge, and Ting Hou, *Member, IEEE*

***Abstract*— To guide the construction of large-scale offshore wind farms, optimization for substation siting and connection topology are both necessary, which is a multiobjective optimization problem. Non-iterative methods are based on greedy strategies and they are only suitable to optimize the connection topology. Iterative methods can update the solutions iteratively to approach the optimum using common optimizers such as particle swarm and firefly algorithm (FA), which are more adaptive in multiobjective optimization. Thus, it is feasible to explore iterative methods to synchronously optimize substation siting and connection topology. This paper proposes a modified FA for the optimization of substation siting and connection topology in a large-scale offshore wind farm. The objective function comprehensively considers critical factors including substation siting, partition of wind turbines, connection topology, cable types, and power loss. The optimization ability of the proposed FA is enhanced by adopting reproduction and resetting mechanisms with dynamic hyperparameters. An implementation that bridges the topological space and Euclidean space is detailed to help with improving the convexity and continuity of search spaces. To validate the efficacy, the proposed FA is first tested in an offshore wind farm with a single substation and then it is applied in a large-scale offshore wind farm with multiple substations to demonstrate the synchronous optimization of substation siting and connection topology.**

***Index Terms*— Connection topology, firefly algorithm, offshore wind farm, optimization, substation siting.**

Nomenclature

G	Representation of graphs in adjacency table.
A_G	The adjacency matrix of G.
G_{st}	The spanning tree of graph G.
V, V_{st}	The vertice set of graph G and G_{st}.
E, E_{st}	The edge table of graph G and G_{st}.
W, W_{st}	The weight table of graph G and G_{st}.
W, w	The cost of cables, the cost per unit length is in the lowercase.
p	Type of cable.
$n, n_{\mathrm{os}}, n_{\mathrm{wt}}$	Number of vertices, OSs and WTs in a graph.
$n_{\mathrm{wtsub}\ i}$	Number of vertices in the subtree of G_{st} that WT i is the root.
n_{wtfather}	Number of the father vertices of a WT.
n_{wtchild}	Number of the child vertices of a WT.
$S_{\mathrm{n},p}, U_{\mathrm{n},p}, I_{\mathrm{n},p}$	The rated apparent power, voltage and current of cable type p.
I_{wt}	The rated current of a single WT.
C	A cut-set of graph G.
$\boldsymbol{x}, \mathbb{D}$	The latent representation of one solution and its range.
$\boldsymbol{x}_{\mathrm{st}}, \boldsymbol{x}_{\mathrm{os}}$	The latent representation of cable topology, the coordinates of OSs.
$\mathbb{D}_{\mathrm{st}}, \mathbb{D}_{\mathrm{ex}}$	The range of $\boldsymbol{x}_{\mathrm{st}}$ and the excluded solutions.
$\mathbb{G}_{\mathrm{st}}$	All possibly spanning trees of G.
$f(\boldsymbol{x})$	The objective function.
n_{search}	Number of searches of algorithm.
$n_{\mathrm{pop}}, n_{\mathrm{iter}}$	Population size of algorithm, number of iterations.

I. Introduction

RENEWABLE wind energy has been rapidly developing in recent years [1], [2], [3], [4]. For offshore wind farms, wind power can be exploited with a relatively smaller environmental impact [5], [6]. However, investment and construction of large-scale offshore wind farms with multiple offshore substations (OSs) are still challenging, and the key problems include optimization of OS siting and connection topology, as shown in Fig. 1. Literature shows that cable investment has taken up 9% of total investment in offshore wind farms [5], and it can be substantially reduced with an optimized construction scheme, which is significantly affected by where the OS locates, how the electric cables connect, and what types of the electric cables are used. Moreover, power consumption is a long-term factor that also affects cost-effectiveness. All these critical factors should be comprehensively considered. Unlike the onshore wind farms, the cable connections in offshore scenarios are less constrained by landforms or human activities, which means it is more

Manuscript received 28 February 2023; revised 3 June 2023; accepted 14 June 2023. Date of publication 28 June 2023; date of current version 13 September 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 52007067 and in part by the Natural Science Foundation of Guangdong Province under Grant 2022A1515011581 and Grant 2023A1515011623. This article was recommended by Guest Editor J. Wu. *(Corresponding author: Zhicong Huang.)*

Zhicong Huang, Canjun Yuan, and Hanchen Ge are with the Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: zhiconghuang@scut.edu.cn).

Ting Hou is with the State Key Laboratory of HVDC, Electric Power Research Institute, China Southern Power Grid, Guangzhou 510080, China.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JETCAS.2023.3290161.

Digital Object Identifier 10.1109/JETCAS.2023.3290161

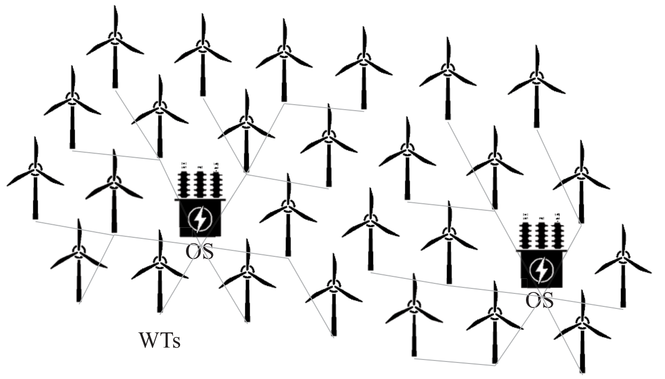


Fig. 1. Typical configuration of offshore wind farms (OS and WT represent offshore substation and wind turbine respectively).

applicable to investigate an optimization algorithm to solve the problem.

In the offshore wind farm, the OS siting and connection topology can be regarded as a minimum spanning tree (MST) problem. To be specific, the weighting factors of branches are variable due to the type selection of cables. For the optimization problem of OS siting and connection topology in this paper, the domain of definition is all possibly spanning trees of a fully-connected graph [7], where every pair of distinct vertices is connected by a unique edge. The complexity exponentially correlates with the scale of the offshore wind farm, and thus a brute force enumeration method cannot be applied. It motivates us to explore an effective algorithm to optimize the OS siting and connection topology in an offshore wind farm.

Some optimization methods for offshore wind farms have been carried out, as shown in Table I, which can usually be categorized into non-iterative and iterative methods. The non-iterative methods build the solution directly according to greedy strategies. And they are mostly modified based on the typical MST algorithms such as Prim method and Kruskal method [8]. In the Prim method, the smallest branch in the cut-set between the result and other vertices is greedily selected at every step and added to the final result. Dynamic minimum spanning tree (DMST) in [9] inherits the greedy strategy of the Prim method, with the consideration of selecting cable types. In the DMST, the branch in the cut-set that brings the smallest increment of the total cost is selected at every step rather than the smallest-weighted branch. However, it should be noted that the MST-based methods are only suitable for optimizing connection topology in offshore wind farms. In [10], to extend the optimization for OS siting, a fuzzy C-means clustering algorithm (FCM) is used in addition to a modified MST algorithm. The FCM is responsible for partitioning wind turbines (WTs) and setting the center of each partition as the OS location, while afterward, the MST optimizes the connection topology. However, this two-step optimization method may miss out on the optimum solutions since the partition of WTs and the connection topology optimization are deeply coupled. For the non-iterative methods, the greedy choices for the smallest cost increment in each step do not mean the global optimum under certain circumstances, and the global optimum may not be found due to missing backtracking strategies.

Alternatively, iterative method is another kind of method for this problem which iteratively approaches the global optimum using common optimizers, e.g. stochastic gradient descent (SGD), genetic algorithm (GA) [11], particle swarm optimization (PSO), firefly algorithm (FA), etc. In these optimizers, one or several solutions are first initialized in the search space, and the solutions are then updated step by step according to the objective function and boundary conditions. Different updating schemes are adopted in these methods, e.g. in SGD [12], only one solution is introduced, and the gradient of the objective function determines the step size and direction. PSO and FA abandoned the dependency of gradients and introduced a group of solutions updating respectively. In PSO [13], the update of solutions depends on the historical best of each individual and the entire group, and in FA [14], the solutions are attracted by each other, with their attractiveness proportional to the objective function. Generally, SGD is appropriate for derivable objective functions, PSO is easy to be implemented and fine-tuned, and FA is more capable of local searches and converges faster. In the optimization problem for offshore wind farms specifically, the solutions of cable connection topology and substation siting have to be transformed into Euclidean space vectors before they can be updated by optimizers. Therefore, various transformation schemes are proposed [6], [15], [16], [17], [18], [19], and different optimizers are selected based on the transformed solution space characteristics.

Qi combined the Voronoi diagram to improve the partition process of the offshore wind farm and used Q-learning to optimize the hyperparameters of the PSO to obtain better optimization results [20]. Yuan proposed an improved partheno genetic algorithm to optimize the topology of the large-scale offshore wind farm collection system and studied the impact of voltage level on the topology optimization of the collection system [21]. However, the current studies of iterative methods for synchronous optimization of OS siting and connection topology are not thorough enough, and the transformed search spaces in these works are not smooth enough to make the optimizers converge stably. Therefore a considerable gap between the global optimal and the final solution remains.

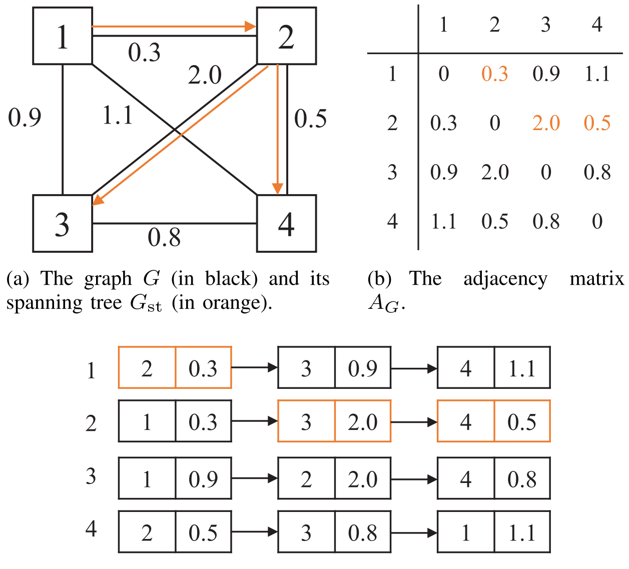
To this end, there are still challenges in optimizing OS siting and connection topology in offshore wind farms with multiple OSs. The non-iterative methods are only suitable for optimizing the connection topology and the greedy strategies they take are incorrect under some circumstances. For iterative methods, their precision is yet to be improved, and the synchronous optimization of both OS siting and connection topology is missing. To guide the construction of a large-scale offshore wind farm with multiple OSs, a modified FA method has been developed to optimize OS siting and connection topology, and the major novelties and contributions are summarized as follows.

1) Synchronous optimization of OS siting and connection topology is enabled for the large-scale offshore wind farm, and critical factors including OS siting, WTs

TABLE I

An Overview of Optimization Methods for Cable Configuration of Offshore Wind Farms

Category	Algorithm	References	Cable topology	Cable types	Power loss	OS Locating	Multiple OS	Synchronous
Non-iterative	Prim	[8]	√					
	DMST	[9]	√	√				
	STMST	[10]	√	√	√	√	√	
Iterative	GA	[6]	√	√			√	
	GA	[15]	√	√	√			
	PSO	[16]	√	√				
	PSO	[17]	√	√	√	√		
	FA	[18]	√	√				
	FA	[19]	√	√				
	Modified FA	this paper	√	√	√	√	√	√



(a) The graph G (in black) and its spanning tree G_{st} (in orange).

	1	2	3	4
1	0	0.3	0.9	1.1
2	0.3	0	2.0	0.5
3	0.9	2.0	0	0.8
4	1.1	0.5	0.8	0

(b) The adjacency matrix A_G.

1	2	0.3	3	0.9	4	1.1
2	1	0.3	3	2.0	4	0.5
3	1	0.9	2	2.0	4	0.8
4	2	0.5	3	0.8	1	1.1

(c) The adjacency table of G and G_{st}.

Fig. 2. An example of graph and spanning tree.

partition, cable topology, cable types, and power loss are comprehensively considered.

2) Optimization ability of the proposed FA method is enhanced by adopting reproduction and resetting mechanisms with dynamic hyperparameters.
3) To help with improving the convexity and continuity of search spaces for the modified FA method, implementation to bridge the spanning tree and the location of OS to the Euclidean space are detailed.

To demonstrate the improvement in performance, the modified FA method is first tested in an offshore wind farm with a single OS. Then, it is applied in a large-scale offshore wind farm with multiple OSs to validate the synchronous optimization of OS siting and connection topology.

The rest of this paper is organized as follows: Section II presents the mathematical model of offshore wind farms and introduces the objective function with boundary conditions. Section III first bridges the spanning tree and the Euclidean space, and then details the proposed modified FA method. The optimization framework is presented. Section IV carries out the experiments to evaluate the proposed method in both offshore wind farms with single and multiple OSs. Finally, Section V concludes this paper.

II. Mathematical Model

A. Mathmatical Representation of Graphs and Spanning Trees

The offshore wind farms can be represented as a fully-connected undirected weighted graphs, given by $G = (V, E, W)$, in which the vertices V are OSs and WTs, and the edges E are the possible connections with their weight W as the cost of cable. Meanwhile, a cable connection scheme can be represented as a spanning tree G_{st}, which is a directed subgraph of G. In spanning trees, the root is OS and each WT is pointed by only one another vertex.

In graph theory, G can be represented as either adjacency matrices or adjacency tables. A graph as well as its spanning tree shown in Fig. 2 is taken as an example. Suppose that n is the number of OSs and WTs, the adjacency matrix A_G is a $n \times n$ square symmetry matrix, and its element at ith row and jth column represents the weight between vertex i and j. Alternatively, the adjacency table is easier to handle spanning trees and takes up fewer spaces. It contains a set of vertices V and two arrays of sets E_k and W_k $(k \in V)$ representing edges and weights respectively. In Fig. 2(a), the adjacency table of the spanning tree G_{st} is given by

$$V_{\mathrm{st}} = \{1, 2\} \subset V \tag{1}$$

$$\begin{cases} E_{\mathrm{st}\ 1} &= \{2\} \subset E_1 \\ E_{\mathrm{st}\ 2} &= \{3, 4\} \subset E_2 \end{cases} \tag{2}$$

$$\begin{cases} W_{\mathrm{st}\ 1} &= \{0.3\} \subset W_1 \\ W_{\mathrm{st}\ 2} &= \{2.0, 0.5\} \subset W_2 \end{cases} \tag{3}$$

Two important concepts in graph theory are highlighted as follows.

1) **Cut-set:** A cut $C = (S, T)$ is a partition of the vertices V of a graph $G = (V, E, W)$ into two disjoint subsets S and T. The cut-set of a cut $C = (S, T)$ is the set of edges that have one endpoint in S and the other endpoint in T.
2) **Minimum spanning tree (MST):** The MST of a graph is the spanning tree with the smallest weight. The Prim method is to find out the MST by greedily selecting the minimum edge in the cut-set.

B. Objective Function

The object of this paper is to optimize the cost of cable in offshore wind farms, which means finding out the spanning tree G_{st} and the location of OSs $\boldsymbol{x}_{\mathrm{os}}$ with the lowest cost of cable in the graph G. Therefore, the objective function is to minimize the cost of cable, which includes cable purchase cost W_{pur}, cable laying cost W_{lay} and power loss W_{loss}, and its expression can be given by

$$\min_{G_{\mathrm{st}},\boldsymbol{x}_{\mathrm{os}}} (W_{\mathrm{total}}) = \min_{G_{\mathrm{st}},\boldsymbol{x}_{\mathrm{os}}} (W_{\mathrm{pur}} + W_{\mathrm{lay}} + W_{\mathrm{loss}}). \tag{4}$$

1) In a wind farm with n_{wt} wind-turbines and n_{os} substations, the cable purchase cost W_{pur} is proportional to the length:

$$W_{\mathrm{pur}} = \sum_{i=1}^{n_{\mathrm{wt}}} w_{p(i)} L_i \tag{5}$$

where $p(i)$ is the type of cable i, w_p is the price per unit length of the type p cable, and the length of cable i is L_i. According to [22], w_p can be calculated as

$$w_p = A_p + B_p \times \mathrm{e}^{\frac{C_p S_{\mathrm{n},p}}{10^8}} \tag{6}$$

$$S_{\mathrm{n},p} = \sqrt{3}\, U_{\mathrm{n},p}\ I_{\mathrm{n},p} \tag{7}$$

where A_p, B_p, and C_p are cost constants, $S_{\mathrm{n},p}$, $U_{\mathrm{n},p}$, $I_{\mathrm{n},p}$ is the rated apparent power, voltage and current of cable type p.

2) The cable laying cost W_{lay} is also proportional to the length, given by:

$$W_{\mathrm{lay}} = \sum_{i=1}^{n_{\mathrm{wt}}} w_{\mathrm{lay}} L_i \tag{8}$$

w_{lay} is the cable laying cost per unit length required for cable installation, and its standard is $w_{\mathrm{lay}} = 360$ CNY/m [23].

3) W_{loss} refers to the cost caused by cable power loss in the whole life cycle of the offshore wind farm, and its specific expression is:

$$W_{\mathrm{loss}} = 3T_{\mathrm{Y}} \sum_{i=1}^{n_{\mathrm{wt}}} C_e (n_{\mathrm{wtsub}\ i} I_{\mathrm{wt}})^2 (r_{p(i)} L_i)(8760 c_f) \tag{9}$$

where T_{Y} is the expected life of the wind farm, and it is 25 years; C_e is the cost corresponding to unit energy loss, its value is 0.85 CNY/kW; $n_{\mathrm{wtsub}\ i}$ is the number of vertices in the subtree of G_{st} that WT i is the root; I_{wt} is the rated current of a single WT, its value is 44 A, and all the WTs in the wind farm have the same rated current; r_p is the unit resistance of the p type cable; c_f is the full capacity coefficient of the wind farm in a year, it is 0.3.

C. Boundary Conditions

The boundary conditions define the solution spaces of this optimization problem, including electrical constraints and geometric constraints.

1) First, the current of a cable should be no greater than its rated current, which can be represented as

$$n_{\mathrm{wtsub}\ i} I_{\mathrm{wt}} \leqslant I_{\mathrm{n},p(i)}. \tag{10}$$

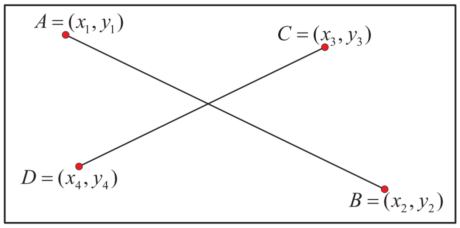


Fig. 3. An example of cable crossing constraint.

2) Second, each WT should have one and only one father vertex, such that the solution is a tree, given by

$$\begin{cases} n_{\mathrm{wtfather}\ i} = 1 \\ n_{\mathrm{wtchild}\ i} \geqslant 0 \end{cases} \tag{11}$$

3) Third, since cross-laying cables will increase installation costs and later maintenance costs, cable crossing should be avoided in the topology design of wind farms, given by

$$E_i \cap E_j = \varnothing \tag{12}$$

where E_i and E_j are two different edges in the spanning tree.
During simulation experiments, matrix calculations can be used to determine whether two cables intersect. Assuming that the four endpoints of two cables in Fig. 3 are A, B, C and D, two specific parameter values can be calculated according to the coordinates of the four endpoints, and the formula is as follows:

$$\begin{bmatrix} x_1 - x_2 & x_4 - x_3 \\ y_1 - y_2 & y_4 - y_3 \end{bmatrix} \cdot \begin{bmatrix} \mu \\ \nu \end{bmatrix} = \begin{bmatrix} x_4 - x_2 \\ y_4 - y_2 \end{bmatrix} \tag{13}$$

If the coefficients μ and ν belong to (0, 1), it means that the cable AB crosses the cable CD, and the intersection is not an endpoint.

III. Optimization Method

A. Transformation of Solution Space

As is illustrated in Section I, meta-heuristic algorithms only work in Euclidean spaces $\mathbb{D} \subset \mathbb{Z}^{n_{\mathrm{wt}}+2n_{\mathrm{os}}}$. To optimize OS siting and connection topology synchronously, a solution in Euclidean space $\boldsymbol{x}$ should include information of both the spanning tree and the location of OS. Thus, the vector in Euclidean space $\boldsymbol{x}$ can be divided into two parts as given by

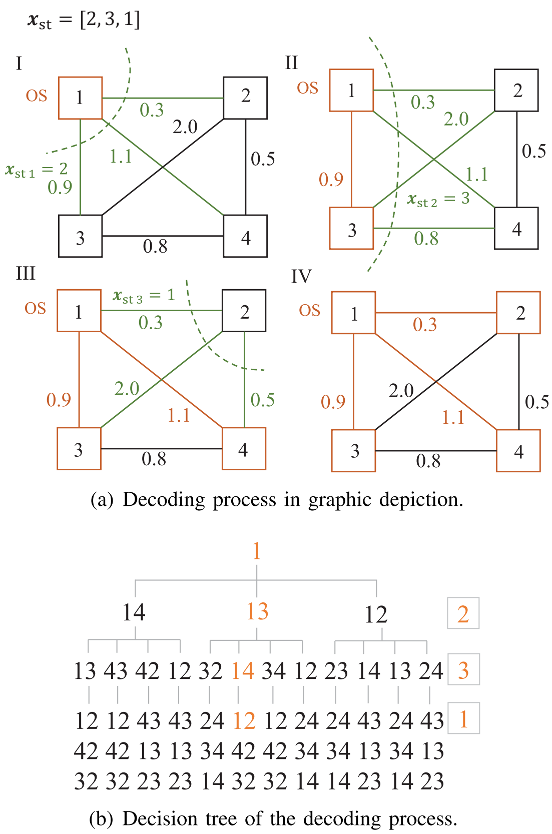
$$\begin{aligned} &\boldsymbol{x} = [\boldsymbol{x}_{\mathrm{os}}, \boldsymbol{x}_{\mathrm{st}}], \\ &\boldsymbol{x}_{\mathrm{os}} \in \mathbb{D}_{\mathrm{os}} \subset \mathbb{Z}^{2n_{\mathrm{os}}}, \boldsymbol{x}_{\mathrm{st}} \in \mathbb{D}_{\mathrm{st}} \subset \mathbb{Z}^{n_{\mathrm{wt}}} \end{aligned} \tag{14}$$

where $\boldsymbol{x}_{\mathrm{os}}$ is the OS coordinates, and $\boldsymbol{x}_{\mathrm{st}}$ is the encoding of cable topology. Therefore, a transformation scheme between the connection topology and Euclidean space vectors should be carried out, given as

$$\mathbb{D}_{\mathrm{st}} \rightleftarrows \mathbb{G}_{\mathrm{st}}, \tag{15}$$

where $\mathbb{G}_{\mathrm{st}}$ is all possibly spanning trees of G.

Such a transformation scheme is crucial to the convexity and continuity of search spaces, which affects the convergence



(a) Decoding process in graphic depiction.

(b) Decision tree of the decoding process.

Fig. 4. Decoding process of a wind farm with single OS.

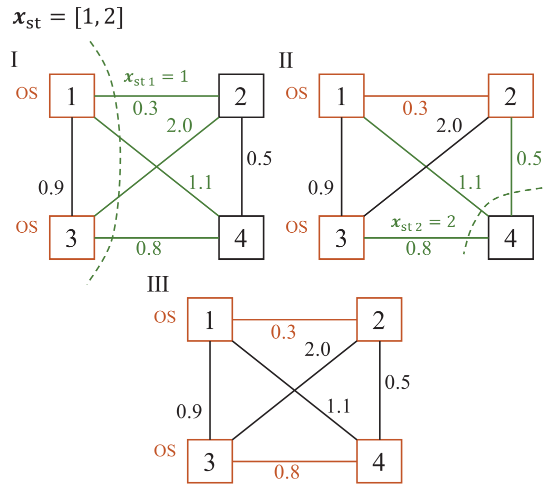


Fig. 5. Decoding process of a wind farm with two OSs.

performance of the algorithm. In this section, a transformation scheme is proposed. Instead of focusing on the structure of the spanning tree itself, the proposed scheme focuses on the order of decisions in the tree's generation process, which corresponds to the greedy strategies in the Prim method. In this method, each element of the encoding $\boldsymbol{x}$ stands for a decision while generating the spanning tree.

The decoding process is illustrated in Algorithm 1. In order to illustrate the decoding method specifically, a simple wind farm G with 3 WTs and 1 OS is shown in Fig. 4, in which a Euclidean space vector $\boldsymbol{x}_{\mathrm{st}} = [2, 3, 1]$ is to be decoded into a spanning tree G_{st}. First, the OS vertice is selected as the root of G_{st}. Then $\boldsymbol{x}_{\mathrm{st}}$ is traversed with G_{st} generated step by step. For the ith step, the cut set between the current spanning tree $G_{\mathrm{st}\,i}$ and the rest of G is selected, which can be depicted by

$$C(G_{\mathrm{st}\,i}, G - G_{\mathrm{st}\,i}), \tag{16}$$

and the $\boldsymbol{x}_i$th smallest edges in this cut set is add into $G_{\mathrm{st}\,i}$. This process has been detailed in Fig. 4(a), where the orange represents G_{st}, the green edges are the cut-set C. Finally a complete G_{st} is generated.

Detailed analyses of the proposed transformation scheme are illustrated as follows:

1) The mapping relationship between spanning trees and search space vectors in the proposed transformation scheme is evaluated. First, this scheme can decode all possible spanning trees, that is, all spanning trees have their corresponding decision variables, and no spanning trees will be omitted. In the graph example of Fig. 4, there are 3 × 4 × 3 = 36 different encodings and 16 different spanning trees, and the spanning tree encoding [2, 3, 1] is also equivalent to [3, 4, 1], [1, 3, 2], etc. Morover, the spanning tree that a vector represents is unique, since the proposed transformation scheme has ensured that any edges will not be chosen repeatedly and the edges that are not connected to the spanning tree will not be selected. To be summarized, a many-to-one onto mapping is created by the proposed method from the search space vectors into spanning trees, which ensures the optimal solution can be transformed into the search space.

2) For an evaluation of the convexity, the boundary of search space in the proposed transformation scheme is then calculated. To be specific, the decision tree of the decoding example is shown in Fig. 4(b), where the choices at each step are listed as branches. It can be noted that the range of the ith element of the search space vector $\boldsymbol{x}_{\mathrm{st}\,i}$ is equal to the number of options in the cut set. Therefore, in a fully-connected graph G, the range of $\boldsymbol{x}_{\mathrm{st}}$ can be calculated by

$$\begin{aligned}\mathbb{D}_{\mathrm{st}} = \{\boldsymbol{x}_{\mathrm{st}} \in \mathbb{Z}^{n_{\mathrm{wt}}} | 1 \le \boldsymbol{x}_{\mathrm{st}\,i} \le \mathrm{size}(C(G_{\mathrm{st}\,i}, G - G_{\mathrm{st}\,i}))\},\\ \mathrm{size}(C(G_{\mathrm{st}\,i}, G - G_{\mathrm{st}\,i})) = i(n - i),\end{aligned} \tag{17}$$

where $G_{\mathrm{st}\,i}$ means the spanning tree with i vertices, n is the number of vertices. Therefore, $\mathbb{D}$ is convex since each of its dimension is convex. However, (12) illustrates that the solutions which has crossing cables $\mathbb{D}_{\mathrm{ex}}$ should be excluded, which breaks the convexity. This problem is solved by mapping the vectors from $\mathbb{D}_{\mathrm{ex}}$ into the legal domain using random strategies, given by

$$\mathbb{D}_{\mathrm{ex}} \xrightarrow{\text{random}} \mathbb{D}_{\mathrm{st}} - \mathbb{D}_{\mathrm{ex}} \tag{18}$$

such that the solution space is convex.

3) The proposed scheme are also capable in the multi OS senerios, where the partition of WTs can be represented in a single Euclidean space vector. In the proposed scheme, multiple spanning trees can be represented in one single vector. For instance, a simple wind farm with 2 OSs and 2 WTs is shown in Fig. 5, in which

```
Algorithm 1 The Decoding Process of Vectors
Input: The vector $x_{st}$ to be decoded, the adjacency table of
       graph $G$, the index of OS vertex.
Output: The adjacency table of the spanning tree $G_{st}$.
The OS vertex is selected as the root of the spanning tree $G_{st}$.
for i ← 1 to length($x_{st}$) do
    The cut-set $C = (G_{st}, G - G_{st})$ between the spanning
    tree and other vertices are selected.
    Find out the $x_{st\ i}$th smallest weight in the cut-set $C$,
    and add it into $G_{st}$.
end for
```

```
Algorithm 2 Conventional Firefly Algorithm
Generate an initial population of $N$ fireflies $x$.
Objective function $f(x)$.
for $t$ ← 1 to MaxGeneration do
    for $i$ ← 1 to $N$ do
        for $j$ ← 1 to $N$ do
            if $f(x^{(j)}) > f(x^{(i)})$ then
                Get a new firefly according to (20).
                if $f(x'^{(i)}) > f(x^{(i)})$ then
                    Replace firefly $x^{(i)}$ with firefly $x'^{(i)}$.
                    Update the global optimal firefly $x_{best}$.
                end if
            end if
        end for
    end for
end for
```

a Euclidean space vector $x_{st} = [1, 2]$ is to be decoded into two spanning trees G_{st}. Similar with the single OS scenerio, the OSs are selected as roots initially, and the vector is traversed with the spanning trees generated step by step. Such that, the connection between the spanning trees are avoided.

B. Modified Firefly Algorithm

In this section, the modified FA method is proposed to further improve the optimization ability of the conventional FA.

The conventional FA is a metaheuristic algorithm inspired by the flashing behavior of fireflies. In a firefly colony, there are n fireflies distributed in the solution spaces with their light intensity proportional to the objective function, as given by

$$I_i \propto f(x^{(i)}) = -W_{total}(x^{(i)}), x^{(i)} \in \mathbb{D} \tag{19}$$

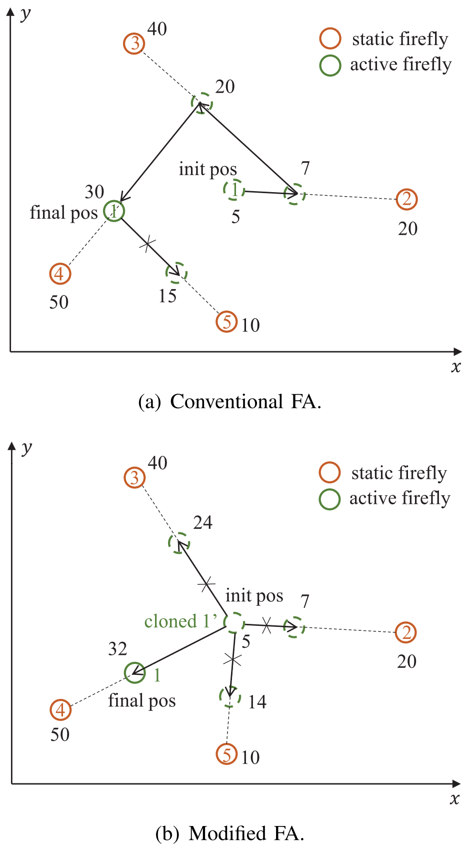
In each iteration, the fireflies are attracted by light and move toward each other, as is given by:

$$x'^{(i)} = x^{(i)} + \beta_0 e^{-\gamma r_{ij}^2}(x^{(j)} - x^{(i)}) + \alpha(\text{rand} - 0.5), \text{ if } I_j > I_i \tag{20}$$

$$r_{ij} = \|x^{(i)} - x^{(j)}\|, \tag{21}$$

$$\text{rand} \sim U(0, 1) \tag{22}$$

```
Algorithm 3 Modified Firefly Algorithm
Generate an initial population of $n_{pop}$ fireflies $x$.
Objective function $f(x)$.
for $t$ ← 1 to $n_{iter}$ do
    Create a new firefly population $y$.
    for $i$ ← 1 to $n_{pop}$ do
        for $j$ ← 1 to $n_{pop}$ do
            if $f(x^{(j)}) > f(x^{(i)})$ then
                Get a new firefly according to (23).
                if $f(y'^{(i)}) > f(y^{(i)})$ then
                    Replace firefly $y^{(i)}$ with firefly $y'^{(i)}$.
                    Update the global optimal firefly $x_{best}$.
                end if
            end if
        end for
    end for
    Sort $x$ and $y$ based on objective function, select new
    firefly population $x$.
end for
```



(a) Conventional FA.

(b) Modified FA.

Fig. 6. A comparison between the conventional FA and modified FA.

(20) is the update formula that the active firefly i move towards each object firefly j respectively, where r_{ij} is the Euclidean distance between them. α is disturbance step size, β_0 is attraction factor, and γ is light absorption coefficient. The second term of (20) represents the attractiveness, and the third term is random noises from the uniform distribution.

To be specific, an example is shown in Fig. 6, which illustrates the movements of an active firefly that lives in a

5-individual colony in the two-dimensional solution space. The orange points are the static object fireflies, and the green points are the motion trail of the active firefly in the current iteration. The black numbers are the objective function value. In the conventional FA shown in Fig. 6(a), the movements of a firefly are accumulated and the objective function value of its destination is pre-estimated every time before it starts. The firefly takes off only if the destination is better.

However, for the optimization of connection topology and OS siting using conventional FA, a considerable gap between the solution and global optimum remains. Firstly, in the search space under the proposed transformation scheme, the distance between two solutions cannot be simply measured using Euclidean distance, thus modifications to the update formula are required. Secondly, the update of a firefly at every iteration depends on the sequence of its objects, which would result in a decrease of stability. Thirdly, a reproduction and elimination mechanism is missing in the conventional FA method.

In view of these problems, three major modifications have been developed to improve the FA, which are summarized as follows, and the graphical example is shown accordingly in Fig. 6(b).

1) The update formula is further improved by introducing dynamic hyperparameters [24], given by

$$\begin{aligned}\boldsymbol{y}'^{(i)} &= \boldsymbol{x}^{(i)} + (\beta_{\min} + (\beta_{\max} - \beta_{\min})e^{-\gamma_t r_{ij}^2})\\ &\quad S_t(\boldsymbol{x}^{(j)} - \boldsymbol{x}^{(i)}) + \alpha_t(\text{rand} - 0.5), \text{ if } I_j > I_i\end{aligned} \tag{23}$$

where the parameters $\alpha_t, \beta, \gamma_t$ are dynamically updated using different strategies including sinusoidal chaotic functions, geometrical annealing, amplitude limit and random disturbance factor S_t etc. Such that, the ability of global search and local search are more balanced and powerful. The specific analytic expression of parameters is as follows.

$$\gamma_t = \frac{t}{M} * \sin(\gamma_t \pi) \tag{24}$$

$$\alpha_t = 0.97^{(400*\frac{t}{M})} \tag{25}$$

$$\beta = (\beta_{\min} + (\beta_{\max} - \beta_{\min})e^{-\gamma_t r_{ij}^2})S_t \tag{26}$$

where t is the current number of iterations, M is the maximum number of iterations and S_t is a random number on the [0,1] interval. The upper limit $\beta_{\max}$ and lower limit $\beta_{\min}$ are determined based on specific problems. After trial and error in the subsequent simulation scenarios, we selected appropriate values. The reason why the disturbance step size is designed in this way is to have $\alpha_t = 0.02$ when the number of iterations reaches half. This can ensure that the algorithm focuses on global search in the first half of the iteration cycle and local search in the second half of the iteration cycle.

2) A resetting mechanism is introduced. In each iteration, fireflies in the modified FA always go back into the initial position first and head off for the next firefly. The firefly is finally ended up at the destination that accounts for the highest objective function value. As is shown in Fig. 6(b), the destination toward the firefly 4 is the best, thus the firefly 1 finally moves toward 4. As a result, the sequence of object fireflies has no impact on the final position of the active firefly in the modified FA, which may help reduce the step sizes and increase stability.

3) A reproduction mechanism is introduced. Fireflies in the modified FA always replicate themselves at the initial position while heading off and get filtered every generation. As is shown in Fig. 6(b), the firefly 1 is ended up with the object function value 32, and its clone 1' is left at the initial position. Therefore, the population of the colony will be doubled at the end of this generation. The colony is then ranked by objective function value and the last half is eliminated. Such a mechanism may introduce competency between fireflies and improve precision.

To be summarized, the processes of conventional FA and modified FA are shown in Algorithm 2 and Algorithm 3 respectively.

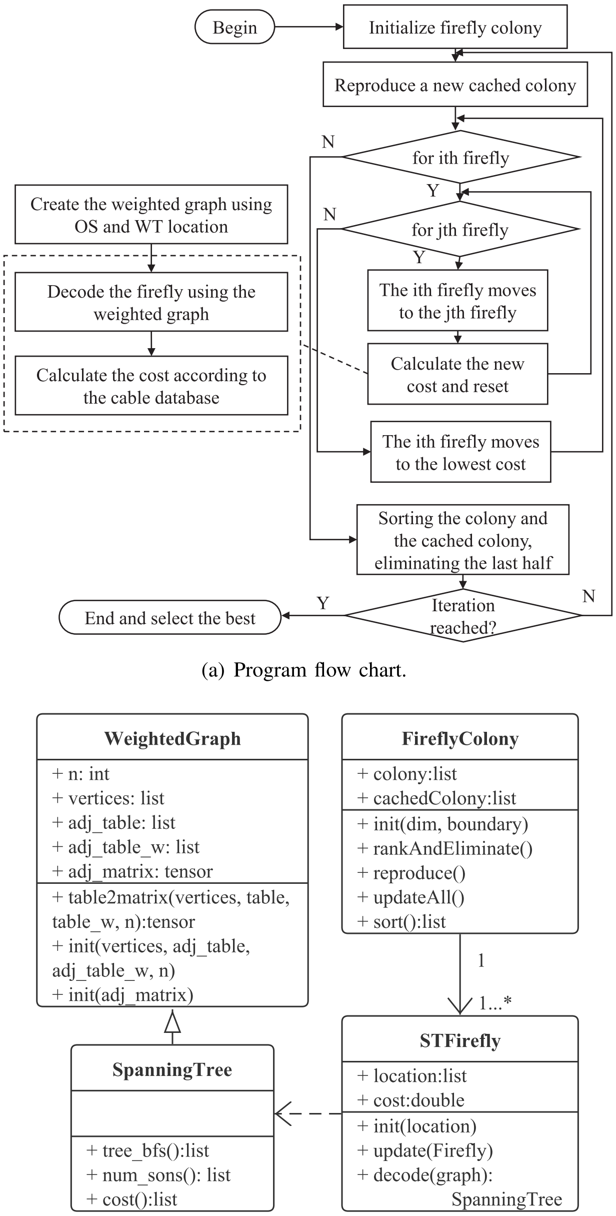
C. Optimization Framework

The structure of the overall optimization framework is shown in Fig. 7 using a program flow chart and UML class diagram respectively, in which the former describes the simulation logics and the latter defines the data structures. The main process of the flow chart as follows:

1) Each firefly will move towards all other individuals that are brighter than itself during its search. After moving several times, each firefly will receive the brightest new individual, which will be cached.
2) Each iteration will perform a double loop operation, and then there will be an original colony and a cache colony. Sort the individuals of these two colonies based on their cost, and select the n_{pop} individuals with the lowest cost to form a new colony for use in the next iteration. The search will end when the maximum number of iterations is reached.
3) When sorting individuals, the spanning tree structure is first decoded based on each individual's decision variable, and then the cost of the spanning tree is calculated using the objective function of (4).

As is shown in Fig. 7(b), 4 crucial components are included: the firefly colony, fireflies for spanning trees, weighted graphs and the spanning trees. Detailed explanations are as follows:

1) *WeightedGraph* is the standard data structure for the topology of offshore wind farms as is illustrated in Section II-A, in which the adjacent table and matrix are included, together with the initializer and conversion method.
2) *SpanningTree* is the subclass of WeightedGraph and represents a detailed connection scheme. The cable type selection and the cost calculation are included.
3) *STFirefly* is the single firefly in the solution space of the Modified FA. It can be decoded into spanning trees and calculates the cost. The update formula of the Modified FA is also included in the member methods.
4) *FireflyColony* is the colony of fireflies that controls the colony to reproduce, update and eliminate. The


Begin
Initialize firefly colony
Reproduce a new cached colony
for ith firefly
N
Y
for jth firefly
N
Y
The ith firefly moves to the jth firefly
Calculate the new cost and reset
The ith firefly moves to the lowest cost
Sorting the colony and the cached colony, eliminating the last half
Iteration reached?
Y
N
End and select the best
Create the weighted graph using OS and WT location
Decode the firefly using the weighted graph
Calculate the cost according to the cable database
WeightedGraph
+ n: int
+ vertices: list
+ adj_table: list
+ adj_table_w: list
+ adj_matrix: tensor
+ table2matrix(vertices, table, table_w, n):tensor
+ init(vertices, adj_table, adj_table_w, n)
+ init(adj_matrix)
FireflyColony
+ colony:list
+ cachedColony:list
+ init(dim, boundary)
+ rankAndEliminate()
+ reproduce()
+ updateAll()
+ sort():list
1
1...*
STFirefly
+ location:list
+ cost:double
+ init(location)
+ update(Firefly)
+ decode(graph): SpanningTree
SpanningTree
+ tree_bfs():list
+ num_sons(): list
+ cost():list

(a) Program flow chart.

(b) UML class diagram.

Fig. 7. Overall optimization framework.

initializer for all the fireflies and the monitor of the optimization process are also included.

Since a virtual firefly colony is defined, the movements of the fireflies are finally simulated according to Fig. 7(a). For every generation, each firefly reproduces itself at the starting point, moves towards the other fireflies, resets and finally ends up at the best destinations. At the end of the generation, the colony is ranked by objective function value and the last half is eliminated.

IV. Experimental Evaluation

Based on the optimization algorithms and framework introduced in Section III, we will demonstrate their feasibility and performance through simulation experiments. We conduct our validation experiments using MATLAB 2021b based on i9-12900K CPU hardware support. For the first batch of experiments, we generated a wind farm with 40 WTs using random number seeds and equipped it with 1 OS. The second batch of experiments also generated a wind farm with 80 WTs using random number seeds and equipped it with 2 OSs. We repeated each group of experiments in each batch 10 times and chose the best one as the final result of this method. The capacity of a single WT is 5 MW and the working voltage is 66 kV. The selectable types of 66 kV cables and their power parameters are shown in Table II.

TABLE II
Power Parameters of 66 kV AC Cable

Sectional area (mm^2)	Rated current (A)	Unit price (CNY/m)	Unit resistance (10^{-4} Ω/m)
70	260	1190	2.5
95	320	1264	1.8
150	400	1421	1.2
240	500	1694	0.7

A. Single OS Scenarios

First, the experiment is carried out in wind farm 1 consisting of 1 OS and 40 WTs to investigate whether the modified FA algorithm has improved performance compared to conventional algorithms under the same optimization framework. In wind farm 1, the location of WTs is randomly selected using random number seeds on a grid with the unit size 100m × 100m and a rectangular boundary 12km × 10km.

In addition, three optimizers are respectively implemented and evaluated on the Matlab platform, including PSO, the conventional FA and the modified FA. The detailed hyperparameter settings of the 3 optimizers are fine-tuned for the best performance. PSO uses time-varying inertia weight and time-varying acceleration coefficients [25] to update parameters, conventional FA sets its attraction factor β_0 to 1.5, while the $\beta_{\min}$ and $\beta_{\max}$ of the modified FA's attraction factor are 1.0 and 2.0.

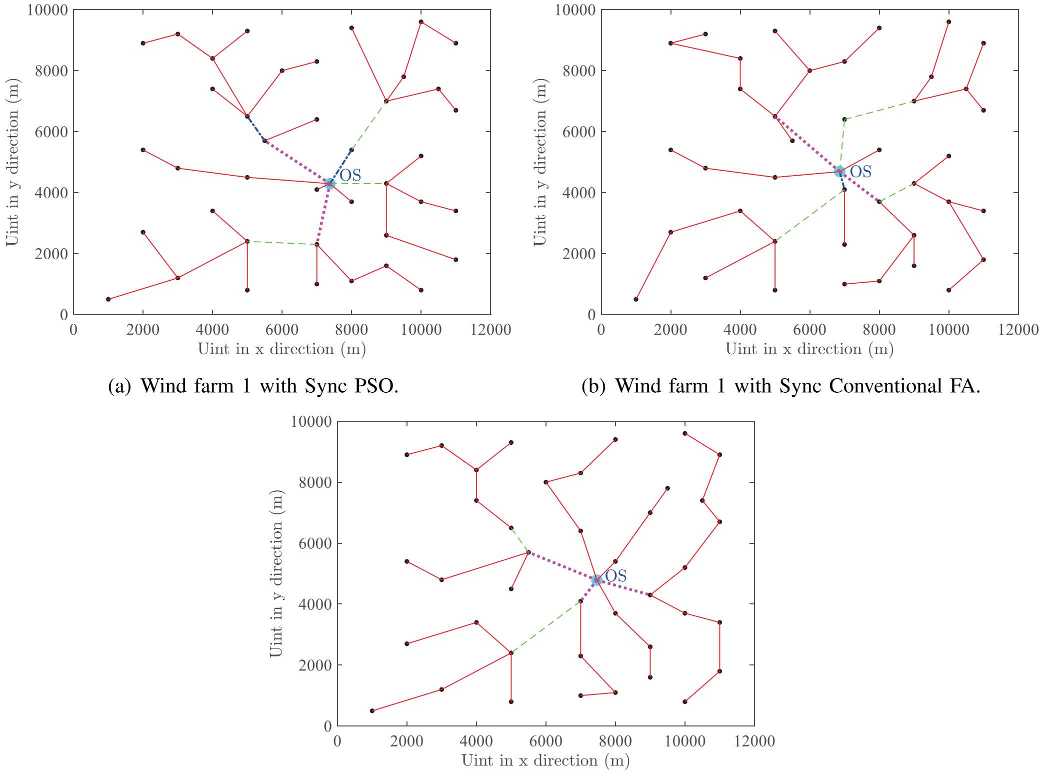
In order to make the comparison between different algorithms fair enough, we defined the number of searches n_{search} to measure the algorithms. PSO belongs to the single cycle optimization algorithm, so its n_{search} is equal to $n_{\text{pop}} \times n_{\text{iter}}$, FA belongs to the double cycle optimization algorithm, so its n_{search} is equal to $n_{\text{pop}}^2 \times n_{\text{iter}}$. The n_{search} is set as basically equal in each pair of experiments.

When decoding decision variables in Euclidean space into a spanning tree, the optimization framework will simultaneously record the length of each selected cable and the number of wind turbines it converges on. Based on the rated current of a single WT, the current flowing in each branch can be calculated, and the appropriate type of cable can be selected by referring to Table II. Using (5), (8) and (9), the three costs of a wind farm can be calculated, i.e. W_{pur}, W_{lay}, and W_{loss}. Then, the total cost of a wind farm W_{total} can be calculated based on (4). In the final optimal solution diagrams, we use four different colors and linestyles to represent different types of cables.

The final experiment results of wind farm 1 are shown in Table III and Fig. 8. Experiment results show that the modified

TABLE III
COMPARISON OF THE SOLUTIONS OF DIFFERENT METHODS FOR WIND FARM 1

Methods	n_{os}	n_{wt}	n_{pop}	n_{iter}	W_{pur} (MCNY)	W_{lay} (MCNY)	W_{loss} (MCNY)	W_{total} (MCNY)	OS coordinates(m)
Sync PSO	1	40	500	500	76.4	22.2	43.8	142.4	(7377, 4292)
Sync Conventional FA	1	40	50	100	76.5	22.3	46.6	145.4	(6859, 4688)
Sync Modified FA	1	40	50	100	72.5	21.1	43.2	136.8	(7466, 4788)



(a) Wind farm 1 with Sync PSO.

(b) Wind farm 1 with Sync Conventional FA.

(c) Wind farm 1 with Sync Modified FA.

Fig. 8. Comparison of optimal solution diagrams of different methods for wind farm 1 (red: 70 mm^2, green: 95 mm^2, blue: 150 mm^2, magenta: 240 mm^2).

FA is the best in terms of purchase cost, laying cost, and power loss cost, and the total cost of the modified FA-based solution has been reduced by 3.93% and 5.91% compared with PSO and the conventional FA, which indicates a closer step into the global optimum and an improvement of precision in this scenario.

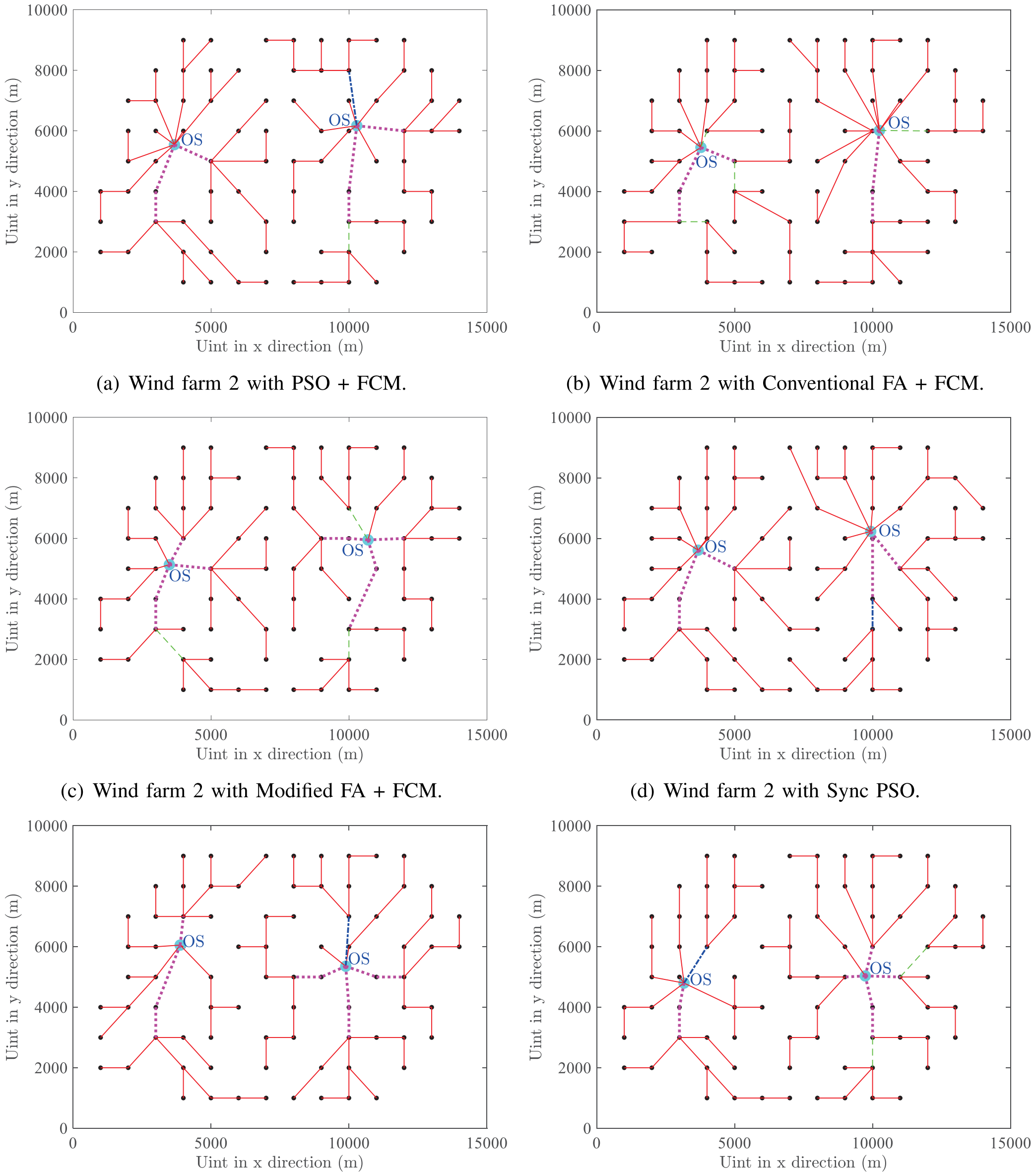
B. Multiple OS Scenarios

Then, the experiment is carried out in the larger wind farm 2 consisting of 2 OSs and 80 WTs to investigate which method performs best under different combinations of algorithms and frameworks. In wind farm 2, the location of WTs is selected randomly using random number seeds on a grid with the unit size 1km × 1km and a rectangular boundary 15km × 10km.

Three optimizers including PSO, the conventional FA and the modified FA are also evaluated respectively in wind farm 2, and the hyperparameter setting is the same as the experiment for wind farm 1. The conventional framework using FCM and the synchronous optimization framework are both applied. In the former framework, FCM partition is first used in which OS siting and connection topology are optimized respectively. In this case, many possible topological solutions are abandoned, which reduced the size of the topological space and was obviously not conducive to the search of subsequent optimization algorithms. In the latter, an end-to-end strategy is applied and all the optimization factors are optimized, which ensures that the topological space is not reduced.

The final experiment results of wind farm 2 are shown in Fig. 9 and Table IV. The final total cost of the modified FA + FCM is 206.9 MCNY, which is cheaper than the 213.7 MCNY of PSO and 223.4 MCNY of conventional FA + FCM, and the total cost is reduced by 3.18% and 7.39%. The total cost of the synchronous modified FA's solution is 205.5 MCNY, which is 3.83% and 8.01% cheaper than that of PSO + FCM and conventional FA + FCM, and is 5.95% and 2.33% cheaper than that of synchronous PSO and synchronous conventional FA respectively. The cost reduction verifies the applicability and accuracy of the proposed algorithm and framework in the multi-OS scenarios.

According to the experimental results above, the proposed modified FA algorithm with a synchronous framework achieves better performance than the conventional methods,



(a) Wind farm 2 with PSO + FCM.

(b) Wind farm 2 with Conventional FA + FCM.

(c) Wind farm 2 with Modified FA + FCM.

(d) Wind farm 2 with Sync PSO.

(e) Wind farm 2 with Sync Conventional FA.

(f) Wind farm 2 with Sync Modified FA.

Fig. 9. Comparison of optimal solution diagrams of different methods for wind farm 2 (red: 70 mm^2, green: 95 mm^2, blue: 150 mm^2, magenta: 240 mm^2).

TABLE IV
Comparison of The Solutions of Different Methods for Wind Farm 2

Methods	n_{os}	n_{wt}	n_{pop}	n_{iter}	W_{pur} (MCNY)	W_{lay} (MCNY)	W_{loss} (MCNY)	W_{total} (MCNY)	OS coordinates(m)
PSO + FCM	2	80	500	500	118.5	34.3	60.9	213.7	(3677, 5547); (10285, 6172)
Conventional FA + FCM	2	80	50	100	123.1	36.0	64.3	223.4	(3779, 5455); (10249, 6015)
Modified FA + FCM	2	80	50	100	112.8	32.4	61.7	206.9	(3492, 5132); (10691, 5951)
Sync PSO	2	80	1000	1920	121.5	35.5	61.5	218.5	(3678, 5612); (9932, 6231)
Sync Conventional FA	2	80	80	300	111.2	31.9	67.3	210.4	(3883, 6054); (9884, 5355)
Sync Modified FA	2	80	80	300	109.0	31.8	64.7	205.5	(3166, 4799); (9734, 5035)

which can further reduce the construction cost of offshore wind farms.

V. Conclusion

A modified FA for the synchronous optimization of OS siting and connection topology in a large-scale offshore wind farm is proposed. Critical factors including OS siting, partition of wind turbines, connection topology, cable types, and power loss are comprehensively considered in the objective function. Compared with conventional FA, by adopting reproduction and resetting mechanisms with dynamic hyperparameters, the optimization ability of the proposed FA is enhanced. To enable synchronous optimization, implementation that bridges the topological space and Euclidean space is detailed to help with improving the convexity and continuity of search spaces. The proposed FA is first tested in an offshore wind farm with a single OS and then it is applied in a large-scale offshore wind farm with multiple OSs to demonstrate the capability of synchronous optimization.

References

[1] H. Díaz and C. G. Soares, "Review of the current status, technology and future trends of offshore wind farms," *Ocean Eng.*, vol. 209, Aug. 2020, Art. no. 107381.

[2] S. Dutta and T. J. Overbye, "Optimal wind farm collector system topology design considering total trenching length," *IEEE Trans. Sustain. Energy*, vol. 3, no. 3, pp. 339–348, Jul. 2012.

[3] R. Perveen, N. Kishor, and S. R. Mohanty, "Off-shore wind farm development: Present status and challenges," *Renew. Sustain. Energy Rev.*, vol. 29, pp. 780–792, Jan. 2014.

[4] M. F. Howland, S. K. Lele, and J. O. Dabiri, "Wind farm power optimization through wake steering," *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 29, pp. 14495–14500, Jul. 2019.

[5] P. Hou, J. Zhu, K. Ma, G. Yang, W. Hu, and Z. Chen, "A review of offshore wind farm layout optimization and electrical system design methods," *J. Modern Power Syst. Clean Energy*, vol. 7, no. 5, pp. 975–986, Sep. 2019.

[6] F. M. Gonzalez-Longatt, P. Wall, P. Regulski, and V. Terzija, "Optimal electric network design for a large offshore wind farm based on a modified genetic algorithm approach," *IEEE Syst. J.*, vol. 6, no. 1, pp. 164–172, Mar. 2012.

[7] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proc. Amer. Math. Soc.*, vol. 7, no. 1, pp. 48–50, Jan. 1956.

[8] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Syst. Tech. J.*, vol. 36, no. 6, pp. 1389–1401, Nov. 1957.

[9] P. Hou, W. Hu, and Z. Chen, "Offshore wind farm cable connection configuration optimization using dynamic minimum spanning tree algorithm," in *Proc. 50th Int. Universities Power Eng. Conf. (UPEC)*, Sep. 2015, pp. 1–6.

[10] T. Zuo, K. Meng, Z. Tong, Y. Tang, and Z. Dong, "Offshore wind farm collector system layout optimization based on self-tracking minimum spanning tree," *Int. Trans. Electr. Energy Syst.*, vol. 29, no. 2, Feb. 2019, Art. no. e2729.

[11] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, pp. 66–73, 1992.

[12] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, Sep. 1951.

[13] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. ICNN International Conf. Neural Netw.* Perth, WA, Australia, Nov. 1995, pp. 1942–1948.

[14] X. S. Yang, "Firefly algorithms for multimodal optimization," in *Proc. Stochastic Algorithms, Found. Appl., 5th Int. Symp.*, 2009, pp. 169–178.

[15] A. M. J. Pemberton, T. D. Daly, and N. Ertugrul, "On-shore wind farm cable network optimisation utilising a multiobjective genetic algorithm," *Wind Eng.*, vol. 37, no. 6, pp. 659–673, Dec. 2013.

[16] P. Hou, W. Hu, and Z. Chen, "Optimisation for offshore wind farm cable connection layout using adaptive particle swarm optimisation minimum spanning tree method," *IET Renew. Power Gener.*, vol. 10, no. 5, pp. 694–702, May 2016.

[17] R. Jin, P. Hou, G. Yang, Y. Qi, C. Chen, and Z. Chen, "Cable routing optimization for offshore wind power plants via wind scenarios considering power loss cost model," *Appl. Energy*, vol. 254, Nov. 2019, Art. no. 113719.

[18] R. Srikakulapu and U. Vinatha, "Combined approach of firefly algorithm with travelling salesmen problem for optimal design of offshore wind farm," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2017, pp. 1–5.

[19] C. El Mokhi and A. Addaim, "Optimal design of the cable layout in offshore wind farms using firefly algorithm and minimum spanning tree," in *Proc. 7th Int. Conf. Optim. Appl. (ICOA)*, May 2021, pp. 1–5.

[20] Y. Qi, P. Hou, and R. Jin, "Optimization of electrical system topology for offshore wind farm based on Q-learning particle swarm optimization algorithm," *Autom. Electr. Power Syst.*, vol. 45, no. 21, pp. 66–75, Nov. 2021.

[21] Y. Yuan, X. Chen, H. Huang, R. Ye, J. Yang, and J. Li, "Topology optimization of large-scale offshore wind power collection system based on improved partheno genetic algorithm," *Water Resour. Power*, vol. 41, no. 1, pp. 212–216, Jan. 2023.

[22] S. Lundberg, "Performance comparison of wind park configurations," Dept. Electr. Power Eng., Chalmers Univ. Technol., Goteborg, Sweden, Tech. Rep., Jul. 2003.

[23] J. Green, A. Bowen, L. J. Fingersh, and Y.-H. Wan, "Electrical collection and transmission systems for offshore wind power," in *Proc. All Days*, Apr. 2007, pp. 1–10.

[24] A. H. Gandomi, X.-S. Yang, S. Talatahari, and A. H. Alavi, "Firefly algorithm with chaos," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 1, pp. 89–98, Jan. 2013.

[25] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 240–255, Jun. 2004.

Zhicong Huang (Member, IEEE) received the B.Eng. degree in electrical engineering and automation and the M.Eng. degree in mechanical and electronic engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2010 and 2013, respectively, and the Ph.D. degree in power electronics from The Hong Kong Polytechnic University, Hong Kong, in 2018.

He is currently an Associate Professor with the Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China. His current research focuses on power electronics techniques in electric vehicles and power systems.

Dr. Huang received the Outstanding Reviewer Award from IEEE TRANSACTIONS ON POWER ELECTRONICS in 2021.

Canjun Yuan was born in Huanggang, China, in 2000. He received the B.Eng. degree in automation from the Wuhan University of Science and Technology, Wuhan, China, in 2022. He is currently pursuing the M.Eng. degree in electronic information with the Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China. His current research interests include machine learning applied in power electronics.

Hanchen Ge was born in Huangshan, China, in 1999. He received the B.Eng. degree in electrical engineering and automation from North China Electric Power University, Beijing, China, in 2021. He is currently pursuing the M.Eng. degree in intelligent engineering with the Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China. His current research interests include artificial intelligence techniques applied in power electronics.

Ting Hou (Member, IEEE) received the M.Eng. degree in power electronics from the Huazhong University of Science and Technology, Wuhan, China, in 2007. She is currently with the State Key Laboratory of HVDC, Electric Power Research Institute, China Southern Power Grid, Guangzhou, China. Her research interests include HVDC transmission technology and reliability analysis of converter valves.