Research on Multi-objective Active Power Optimization Simulation of Novel Improved Water Cycle Algorithm

Gonggui Chen, Ying Han, Zhizhong Zhang*, Xianjun Zeng and Shuaiyong Li

***Abstract*— Water cycle algorithm (WCA) is a heuristic algorithm proposed in recent years. To overcome the insufficiency of standard WCA algorithm in solving the non-convex optimal power flow (OPF) problems, the multi-objective novel improved water cycle algorithm (MONIWCA) is proposed in this paper. The evaporation process is improved in WCA by introducing evaporation rate and the normal distribution optimization mechanism is used to mutate the individual position. The modified WCA also adopts a constraint-based strategy to ensure zero constraint violations. In order to obtain a high-quality Pareto optimal solution set (POS) and select the best compromise solution (BCs), a global ranking strategy is proposed. The global ranking strategy includes the novel constraint handling method, the rank index calculation and the BCs on fuzzy satisfaction theory to deal with the complex constraints of the optimization problem. The MONIWCA has been simulated under the constraints of zero violations on IEEE 30, IEEE 57 and IEEE 118 standard test systems, including six dual-objective cases and one tri- objective case. The simulation results are compared with the multi-objective particle swarm optimization (MOPSO). The results show that the improved method can effectively solve the MOOPF problem, not only to obtain a uniform continuous Pareto solution set but also to achieve a better compromise solution. In addition, the two performance indicators of the generational distance (GD) and the spacing (SP) also show that the MONIWCA algorithm has uniform distribution, high convergence and strong stability.**

***Index Terms*— multi-objective novel improved water cycle algorithm; optimal power flow; global ranking strategy.**

Manuscript received June 1, 2020; revised January 21, 2021. This work was supported by the National Natural Science Foundation Project of China (No.61703066), Natural Science Foundation Project of Chongqing (No.cstc2018jcyjAX0536), Innovation Team Program of Chongqing Education Committee (CXTDX201601019) and Chongqing University Innovation Team under Grant (KJTD201312).

Gonggui Chen is a professor of Key Laboratory of Industrial Internet of Things & Networked Control, Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: chenggpower@126.com).

Ying Han is a master degree candidate of Chongqing Key Laboratory of Complex Systems and Bionic Control, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: hanying9658@126.com).

Zhizhong Zhang is a professor of Key Laboratory of Communication Network and Testing Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (corresponding author to provide phone: +862362461681; e-mail: zhangzztx@163.com).

Xianjun Zeng is a senior engineer of State Grid Chongqing Electric Power Company, Chongqing 400015, China (e-mail: 13594255525@139.com).

Shuaiyong Li is a professor of Key Laboratory of Industrial Internet of Things & Networked Control, Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: lishuaiyong@cqupt.edu.cn).

I. INTRODUCTION

Electricity is one of the most widely used sources in the world. With the continuous expansion of the power system and the rise of power market operations, it is necessary to simultaneously consider the optimization of multiple contradictory objectives and coordinate the competition among multi-objective optimization.

The OPF is designed to make the system safer and more economical while ensuring that the constraints are not violated. Multi-objective optimal power flow (MOOPF) is the problem of simultaneously optimizing multiple contradictory objective functions. The MOOPF, which takes multiple competing goals into consideration concurrently, measures the state of the power system more synthetically [1, 2]. Due to the non-convex and non-linear characteristics of MOOPF problems, heuristic algorithms are more suitable compared with traditional methods [3, 4]. The classic weighting method sets different weight coefficients for multiple targets according to the preference of the decision maker. The Pareto method to solve the MOOPF problem is to select the currently suitable compromise from the candidate solution set. So far, some appropriate algorithms like the multi-objective bees algorithm [5], the novel quasi-oppositional modified Jaya algorithm [6], the teaching-learning based optimization [7], the multi-objective electromagnetism-like algorithm [8], the multi-objective improved bat algorithm [9], and the multi-objective particle swarm optimization [5, 10, 11], are successful to solve the MOOPF problem. There have been many articles about the above algorithm to solve the MOOPF problem, but the WCA is the first application in the field of power system active optimization in this paper.

The water cycle algorithm [12-16] is a heuristic algorithm designed according to the natural water cycle phenomenon, which simulates the process of streams and rivers flowing into the sea. So far, there are many applications in different fields of research utilized the efficiency of WCA for solving complex optimization problems [17], such as the optimal operation of reservoir system [18] and antenna array synthesis [19]. This paper introduces the water cycle algorithm to solve the multi-objective active optimization problem. Applying the WCA to solve the MOOPF problem, we modify a traditional constraint strategy and adopt a global sorting strategy to handle the complicated constraints of the optimization problem. Because the standard WCA tends to converge prematurely, the original WCA introduces a rainfall process. In order to conduct second deep search near the search area and increase the population diversity, the normal distribution optimization mechanism can be used to mutate the individual's optimal position during the search process. The algorithm performs simulation experiments on the IEEE 30, IEEE 57 and IEEE 118 standard test systems under the constraint of zero violations. In addition, the GD and SP indicators are used to measure stability and convergence. The results clearly show that when using the same constrained advantage strategy, MONIWCA has better exploration capabilities than ordinary MOPSO in finding the more competitive BCs.

The following sections of this paper are organized as follows: Section II introduces the mathematical model description of the MOOPF problem. Section III introduces the basic water cycle algorithm and its improvement. In order to obtain high-quality POS and filter satisfactory BCs, three multi-objective strategies are adopted to propose MONIWCA to deal with MOOPF. Section IV shows the simulation results and performance analysis of the algorithm, showing that MONIWCA has advantages in obtaining the BCs and has strong stability. Section V provides the final conclusion.

II. Mathematical model

The multi-objective optimization problem (MOP) can be defined as an optimization problem that minimizes two or more objective functions simultaneously, which satisfies the equality and inequality constraints.

$$\min \quad F = (f_1(x,u),\cdots, f_i(x,u),\cdots, f_M(x,u)) \tag{1}$$

Subject to:

$$H_k(x,u)=0,\quad k=1,2,\cdots,E \tag{2}$$

$$G_j(x,u)\le 0,\quad j=1,2,\cdots,I \tag{3}$$

where F represents the ith objective function. H_k is the kth equality constraint and E is the number of equality constraints, G_j is the jth inequality constraint and I is the number of inequality constraints. In the MOOPF problem, x is the vector of state variables and u is the vector of control variables.

A. Objective functions of MOOPF

The MOOPF is optimized by adjusting the vector of control variables: generator active power output P_G, load node voltage V_L, tap rations of transformer T and reactive power injection Q_C. The objective functions involved in this paper consist of: active power loss *Ploss*, basic fuel cost *Fcost*, emission *Em*, fuel cost with value-point loadings *Fcost-vp* and voltage stability index *L_index*.

1) *Ploss* minimization

$$Obj\ 1\colon\quad \min\ F_{Ploss}=\sum_{k=1}^{N_L} g_{k(i,j)}[V_i^2+V_j^2-2V_iV_j\cos(\delta_i-\delta_j)]\ MW \tag{4}$$

where N_L is the total number of branches. $g_{k(i,j)}$ is the conductance of the kth branch which connects node i and node j. V_i and V_j are the voltage magnitude of node i and node j. δ_{ij} represents the voltage angle between node i and node j.

2) *Fcost* minimization

$$Obj\ 2\colon\quad \min\ F_{\text{cost}}=\sum_{i=1}^{N_G}(a_i+b_iP_{Gi}+c_iP_{Gi}^2)\ \$/\text{h} \tag{5}$$

where a_i, b_i, c_i are the fuel cost coefficients of the ith generator. P_{Gi} is the active power of the ith generator. N_G indicates the number of generators.

3) *Em* minimization

$$Obj\ 3\colon\quad \min\ Em=\sum_{i=1}^{N_G}[\alpha_iP_{Gi}^2+\beta_iP_{Gi}+\gamma_i+\eta_i\exp(\lambda_iP_{Gi})]\ ton/h \tag{6}$$

where α_i, β_i, γ_i, η_i and λ_i are the emission coefficients of the ith generator.

4) *Fcost-vp* minimization

$$Obj\ 4\colon\quad \min\ F_{\text{cost-vp}}=\sum_{i=1}^{N_G}(a_i+b_iP_{Gi}+c_iP_{Gi}^2 +\left|d_i\times\sin(e_i\times(P_{Gi}^{\min}-P_{Gi}))\right|)\ \$/\text{h} \tag{7}$$

where d_i and e_i are the fuel cost coefficients of the ith generator.

5) *L-index* minimization

$$Obj\ 5\colon\quad \min\ F_{L-index}=\min(L\text{-}index)=\min[\max(L_j)] \tag{8}$$

$$L_j=\left|1-\sum_{i=1}^{N_{PV}}F_{ji}\frac{V_i}{V_j}\right|,\quad j=1,2,\cdots \text{N}_{PQ} \tag{9}$$

where N_{PV} and N_{PQ} are the numbers of PV nodes and the number of PQ nodes. F_{ji} can be estimated from the Y-bus matrix. V_i and V_j are the voltages of the ith PV node and the voltages of the jth PQ node.

B. Constraints of MOOPF

With respect to the five objective functions mentioned above are minimized in the case of guaranteeing zero violation of various the equality constraints (ECs) and the inequality constraints (ICs).

1) ECs

The equation constraint conditions are composed of the active and reactive power flow equations of the system, which can be expressed as:

$$P_{\text{G}i}-P_{\text{L}i}-V_i\sum_{j=1}^{N_i}V_j(G_{ij}\cos\delta_{ij}+B_{ij}\sin\delta_{ij})=0\quad (i=1,2\cdots N) \tag{10}$$

$$Q_{\text{G}i}-Q_{\text{L}i}-V_i\sum_{j=1}^{N_i}V_j(G_{ij}\sin\delta_{ij}-B_{ij}\cos\delta_{ij})=0\ (i=1,2\cdots N_{\text{PQ}}) \tag{11}$$

where N_i is the number of nodes connected to node i (excluding node i). N is the number of system nodes except the slack node. P_{Li} and Q_{Li} represent the active and reactive power of load node i. G_{ij} and B_{ij} are the mutual conductance and the mutual susceptance.

2) ICs

The inequality constraint conditions define the operational limits of the power system equipment and can be divided into state variable inequality constraints and control variable inequality constraints.

a) ICs on state variables

- voltage V_{Li} at PQ node

$$V_{Li,\min}\le V_{Li}\le V_{Li,\max},\quad i\in N_{PQ} \tag{12}$$

- active power P_{Gref} at slack bus

$$P_{\text{Gref}}^{\min}\le P_{\text{Gref}}\le P_{\text{Gref}}^{\max} \tag{13}$$

- reactive power Q_{Gi} at PQ node

$$Q_{Gi,\min}\le Q_{Gi}\le Q_{Gi,\max},\quad i\in N_{PV} \tag{14}$$

- apparent power S_{ij}

$$S_{ij}\le S_{ij,\max},\quad i\in N_L \tag{15}$$

b) ICs on control variables

- active output P_{Gi} at generator node

$$P_{\text{G}i}^{\min}\le P_{Gi}\le P_{\text{G}i}^{\max}, i\in N_G \tag{16}$$

- generator terminal voltage V_{Gi}

$$V_{Gi,\min}\le V_{Gi}\le V_{Gi,\max},\quad i\in N_{PV} \tag{17}$$

- transformer tap-settings T_i

$$T_{i,\min}\le T_i\le T_{i,\max},\quad i\in N_T \tag{18}$$

- reactive power injection Q_C

$$Q_{ci,\min}\le Q_{ci}\le Q_{ci,\max},\quad i\in N_C \tag{19}$$

where N_T and N_C indicate the number of transformers and compensators.

III. MONIWCA FOR MOOPF PROBLEM

This chapter introduces the improvement method of the WCA and the optimization steps of the improved algorithm in the MOOPF problem. In order to overcome the shortcomings of the standard water cycle algorithm in solving the non-convex MOOPF problem, MONIWCA is proposed. The improved algorithm introduces the normal distribution optimization mechanism to modify original update method and adopts three multi-objective strategies including constraint handling method, rank index calculation and fuzzy satisfaction theory.

A. Overview of the basic WCA

The water cycle algorithm [12] was inspired by the natural water cycle phenomenon in 2012 by Hadi Eskandar. The basic idea of the algorithm: the water cycle algorithm generates the initial population N_{pop} through the process of rainfall, and divides three levels according to the fitness value. The optimal individual is defined as the sea, some suboptimal individuals are defined as rivers, and the rest are defined as streams that will flow into sea or river. Through the iterative process, individual is continuously updated and re-divided, and when the number of iterations reaches the maximum number of iterations, global optimal value is found. The main steps are presented as below.

1) Initialization

The initial population can be represented as a matrix of N_{pop}*D. Where D is the dimension of the control variable. The initial population is defined as:

$$Total\ pop=\begin{bmatrix} Sea \\ River_1 \\ River_2 \\ River_3 \\ \vdots \\ Stream_{Nsr+1} \\ Stream_{Nsr+2} \\ Stream_{Nsr+3} \\ \vdots \\ Stream_{Npop} \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & x_3^1 & \cdots & x_D^1 \\ x_1^2 & x_2^2 & x_3^2 & \cdots & x_D^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{N_{pop}} & x_2^{N_{pop}} & x_3^{N_{pop}} & \cdots & x_D^{N_{pop}} \end{bmatrix} \quad (20)$$

where N_{pop} is the total population. N_{Sr} is the sum of rivers and sea.

The number of streams attracted by sea and rivers is defined as the flow density. The designated streams for each river and sea are calculated using the following equation:

$$NS_n = round\left\{\left|\frac{C_n}{\sum_{n=1}^{Nsr} C_n}\right| \times N_{Stream}\right\},\ n=1,2,\cdots,N_{sr} \quad (21)$$

where NS_n represents the number of streams which flow to the specific rivers and sea [11]. C_n indicates fitness value.

2) Update process

The update formula is given as follows:

a) Streams flow into the river

$$X_{Stream}^{i+1} = X_{Stream}^{i} + rand \times C \times (X_{River}^{i} - X_{Stream}^{i}) \quad (22)$$

b) Streams flow into the sea

$$X_{Stream}^{i+1} = X_{Stream}^{i} + rand \times C \times (X_{Sea}^{i} - X_{Stream}^{i}) \quad (23)$$

c) Rivers flow into the sea

$$X_{River}^{i+1} = X_{River}^{i} + rand \times C \times (X_{Sea}^{i} - X_{Stream}^{i}) \quad (24)$$

where *rand* is a random number between 0 and 1. C is a position update coefficient of between 1 and 2, which is usually taken as 2.

3) Rainfall process

In order to avoid the algorithm falling into the local optimal solution, the diversity of the population is increased. When the rainfall conditions are satisfied, the rain process will produce new individuals. There are two ways to generate a new individual.

a) Evaporation used between sea and streams. Perform the rainfall process near the sea and search for optimal values.

$$\begin{aligned} & if\ \left\|X_{Sea}^{i} - X_{Streams}^{i}\right\| < d_{\max} \quad i=1,2,3,\cdots,N_{sr} \\ & \quad X_{Stream}^{new} = X_{sea} + \sqrt{\mu} \times randn(1,N) \\ & end\ if \end{aligned} \quad (25)$$

b) Evaporation used between sea and rivers. The random generation of new individuals in the problem space increases the diversity of the population.

$$\begin{aligned} & if\ \left\|X_{Sea}^{i} - X_{River}^{i}\right\| < d_{\max} \quad i=1,2,3,\cdots,N_{sr} \\ & \quad X_{Stream}^{new} = LB + rand \times (UB - LB) \\ & end\ if \end{aligned} \quad (26)$$

where *rand* is a random number uniformly distributed between 0 and 1. UB and LB are the upper and lower bounds of the search space. μ is the sea area search range. The smaller the value of μ, the smaller the search range.

$$d_{\max}^{i+1} = d_{\max}^{i} - \frac{d_{\max}^{i}}{\max\ iteration} \quad (27)$$

where the number of d_{max} close to 0 is usually taken as eps=2.2204e-16, which controls the search intensity near the sea position and adaptive reduction with the number of iterations.

B. Multi-objective solution strategy

When dealing with MOPs, there is a conflict among the objective functions, and it is difficult to obtain an absolute optimal solution. For the multi-objective water cycle algorithm (MOWCA) optimization problem, it is impossible to judge the pros and cons of the comparison function value directly. In order to handle MOPs, a global ranking multi-objective water cycle algorithm with optimal retention strategy is proposed in this paper.

1) Constraint handling method

When the power flow calculation reaches the maximum number of iterations, each solution must satisfy the equality constraints, otherwise, the solution obtained is meaningless.

For the individual i, the *ECs* are able to ensure zero violation of the constraint in OPF. As for *ICs*, the control variables constraints u_i can be adjusted as (28).

$$u_i = \begin{cases} u_i^{\max}, & u_i > u_i^{\max} \\ u_i, & u_i^{\min} < u_i < u_i^{\max} \\ u_i^{\min}, & u_i < u_i^{\min} \end{cases} \quad (28)$$

In order to make the solution satisfy the constraints of the state variable inequality, a constraint dominant strategy is proposed. The total amount of constraint violations can be used as a criterion for classifying the stream level. The calculation formula is (29).

$$S_v(u) = \sum_{j=1}^{P} \max(h_k(x,u),0) \quad (29)$$

where p is the number of ICs.

Furthermore, the individuals u_1 and u_2 in the solution set are selected, and the priority is judged by the dominant method according to formula (29). The quality of the flow can be judged by the constraint dominant strategy. The specific dominant strategy is shown in TABLE I.

TABLE I
THE CONSTRAINT DOMINANT STRATEGY

If $S_v(u_1) > S_v(u_2)$	u_1 dominates u_2;
If $S_v(u_1) < S_v(u_2)$	u_2 dominates u_1;
If $S_v(u_1) = S_v(u_2)$:	
If $\forall\, i \in \{1,2,\ldots,M\} \le f_i\,(x,u_2)$ *or* $\exists\, j \in \{1,2,\ldots,M\}\ f_j\,(x,u_1) < f_i(x,u_2)$	
u_1 dominates u_2;	
else	
u_2 dominates u_1;	

where $S_v(u)$ indicates the total violations. The $f_i(x,u)$ represents the ith fitness value.

According to the constraint dominant strategy, all the streams can be divided into n levels: $F_1, F_2, F_3, \ldots, F_n$. The hierarchical value of the flow is *rank* (i). A smaller *rank* value means a higher priority, and the same total constraints means that the stream *rank* values are equal.

2) Rank Index Calculation

All streams are able to be hierarchically divided by the constraint dominant strategy. According to the non-dominated sorting method proposed by Deb[20]. For water flowing with the same *rank* (i), their priority needs to be calculated by the *distance* (i) of the crowding distance. The crowding distance can estimate the density of other solutions around the solution. At the same rank value, the greater crowded distance means the higher priority. For any stream i and stream j, the global rank relationship can be described as TABLE II.

TABLE II
THE GLOBAL RANK METHODS

If rank(i) < *rank*(j)	stream i is better than stream j;
If rank(i) > *rank*(j)	stream j is better than stream i;
If rank(i) = *rank*(j)	
If distance(i) > *distance*(j)	
stream i is better than stream j;	
else	
stream j is better than stream i;	

3) BCs on Fuzzy Satisfaction Theory

In the previous description, the best solution of the standard water cycle algorithm is that the objective function fitness value of this solution is better than other candidate solutions. For a multi-objective problem, a set of non-inferior solutions is usually obtained by taking the above non-inferior ranking strategies. The best compromise solution can be determined by applying fuzzy satisfaction theory based on the Pareto solution set. The satisfaction of the ith for the individual μ can be expressed as formula (30).

$$\mu_i = \begin{cases} 1 & f_i \le f_{i\min} \\ \dfrac{f_{i\max} - f_i}{f_{i\max} - f_{i\min}} & f_{i\min} < f_i < f_{i\max},\ i = 1,2,\ldots M \\ 0 & f_i \ge f_{i\max} \end{cases} \tag{30}$$

where f_{imax} and f_{imin} are the maximum and minimum values of the objective function i. Normalize the solutions in the POS can be defined as (31).

$$Nos_i = \frac{\sum_{i=1}^{M} \mu_i}{\sum_{k=1}^{N_p} \sum_{i=1}^{M} \mu_i}\ ,\ k = 1,2,\ldots N_p \tag{31}$$

where μ indicates that the satisfaction range of the solution is (0,1). $\mu = 0$ means dissatisfaction with the function value, and $\mu = 1$ indicates complete satisfaction.

C. MONIWCA algorithm

1) Add an evaporation process

The concept of rainfall in the standard water cycle algorithm can effectively enhance the algorithm's convergence ability. In nature, although the destination is the sea, not all streams can flow into the sea. Some rivers have a limited number of streams that flow slowly and have large amounts of evaporation. Hence, in the novel algorithm, three types of evaporation are introduced: evaporation used between sea and streams; evaporation used between sea and rivers; evaporation among rivers having a few streams. Based on the number of streams assigned to the river, the evaporation rate can be expressed as:

$$EP = \frac{Sum(NS_n)}{N_{sr} - 1} \times rand \quad n = 2,\cdots,N_{sr} \tag{32}$$

Eq. (32) shows the evaporation process only used for streams and rivers. The value of *EP* can be changed (slightly) at each iteration which also gives stochastic nature to the *EP*. About the evaporation process, the *EP* is not the only evaporation condition to be satisfied. The low quality of solutions has also to be given more chances to move and flow to the other high quality of solutions or to find better regions in terms of better objective function value which is given as follows.

$$\begin{aligned} &\textit{for}\quad i = 2 : N_{sr}\text{ -1} \\ &\qquad \textit{if}\quad (\exp(\text{-}k / \max\_it) < rand)\ \&\ (NS_i < ER) \\ &\qquad\qquad X_{Stream}^{new} = LB + rand \times (UB\text{ - }LB) \\ &\qquad \textit{end} \\ &\textit{end} \end{aligned} \tag{33}$$

where k is the iteration index. In the novel algorithm, at early iterations, the probability of evaporation is high for low-quality solutions. It is decreased as the number of iterations increases.

2) Normal distribution optimization mechanism

In the heuristic algorithm, there is a recognized shortcoming that it is easy to prematurely converge. The WCA is not an exception and suffers from premature convergence in dealing with more complex problems. A deep exploitation of the vicinity of the exploitation areas is the key objective of the second stage [21]. The normal distribution optimization mechanism is used to mutate the optimal position of the individual, which means that the search range of the individual becomes large. It is beneficial to increase the diversity of the population and jump out of possible local extremes. Based on the normal distribution optimization mechanism, the update strategy of streams and rivers can be expressed as formula (34) and formula (35).

a) Stream renewal strategy

$$X_{Stream}^{i}(t+1) = \begin{cases} N(0.5 \times [X_{River}^{i}(t) + X_{Stream}^{i}(t)], \left|X_{River}^{i}(t) - X_{Stream}^{i}(t)\right|), \\ \qquad if\ \ p < 0.3 \\ N(0.5 \times [X_{Sea}^{i}(t) + X_{Stream}^{i}(t)], \left|X_{Sea}^{i}(t) - X_{Stream}^{i}(t)\right|), \\ \qquad if\ 0.3 \le p \le 0.5 \\ N(0.5 \times [X_{Sea}^{i}(t) + X_{River}^{i}(t)], \left|X_{Sea}^{i}(t) - X_{River}^{i}(t)\right|), \\ \qquad if\ 0.5 < p \le 0.7 \\ X_{Stream}^{i} + rand \times C \times (X_{Sea}^{i} - X_{Stream}^{i}), \\ \qquad if\ 0.7 < p \end{cases} \tag{34}$$

where p is a random number between 0 and 1. N represents a normal distribution.

b) River renewal strategy

$$X_{River}^{i}(t+1)=\begin{cases} N(0.5\times[X_{River}^{i}(t)+X_{Stream}^{i}(t)],\left|X_{River}^{i}(t)-X_{Stream}^{i}(t)\right|), \\ \quad if\ \ p<0.3 \\ N(0.5\times[X_{Sea}^{i}(t)+X_{Stream}^{i}(t)],\ \left|X_{Sea}^{i}(t)-X_{Stream}^{i}(t)\right|), \\ \quad if\ 0.3\le p\le 0.5 \\ N(0.5\times\ [X_{Sea}^{i}(t)+X_{River}^{i}(t)],\ \left|X_{Sea}^{i}(t)-X_{River}^{i}(t)\right|), \\ \quad if\ 0.5<p\le 0.7 \\ X_{River}^{i}+rand\times C\times(X_{Sea}^{i}-X_{Stream}^{i}), \\ \quad if\ 0.7<p \end{cases} \tag{35}$$

D. Application of the MONIWCA to the MOOPF problem

In order to verify the superiority of the algorithm, the proposed improved algorithm was applied to the MOOPF problem, and 30 independent experiments were simulated on IEEE 30, IEEE 57 and IEEE 118 systems. Compared with the MOPSO algorithm, bi-objective cases are studied. Detailed experimental results are shown in Section IV. The flow chart of the main steps of the improved multi-objective water cycle algorithm to solve the MOOPF problem is shown in Fig. 1. TABLE III shows the specific combinations of objective functions involved in this article, including 7 cases involving 5 objective functions.

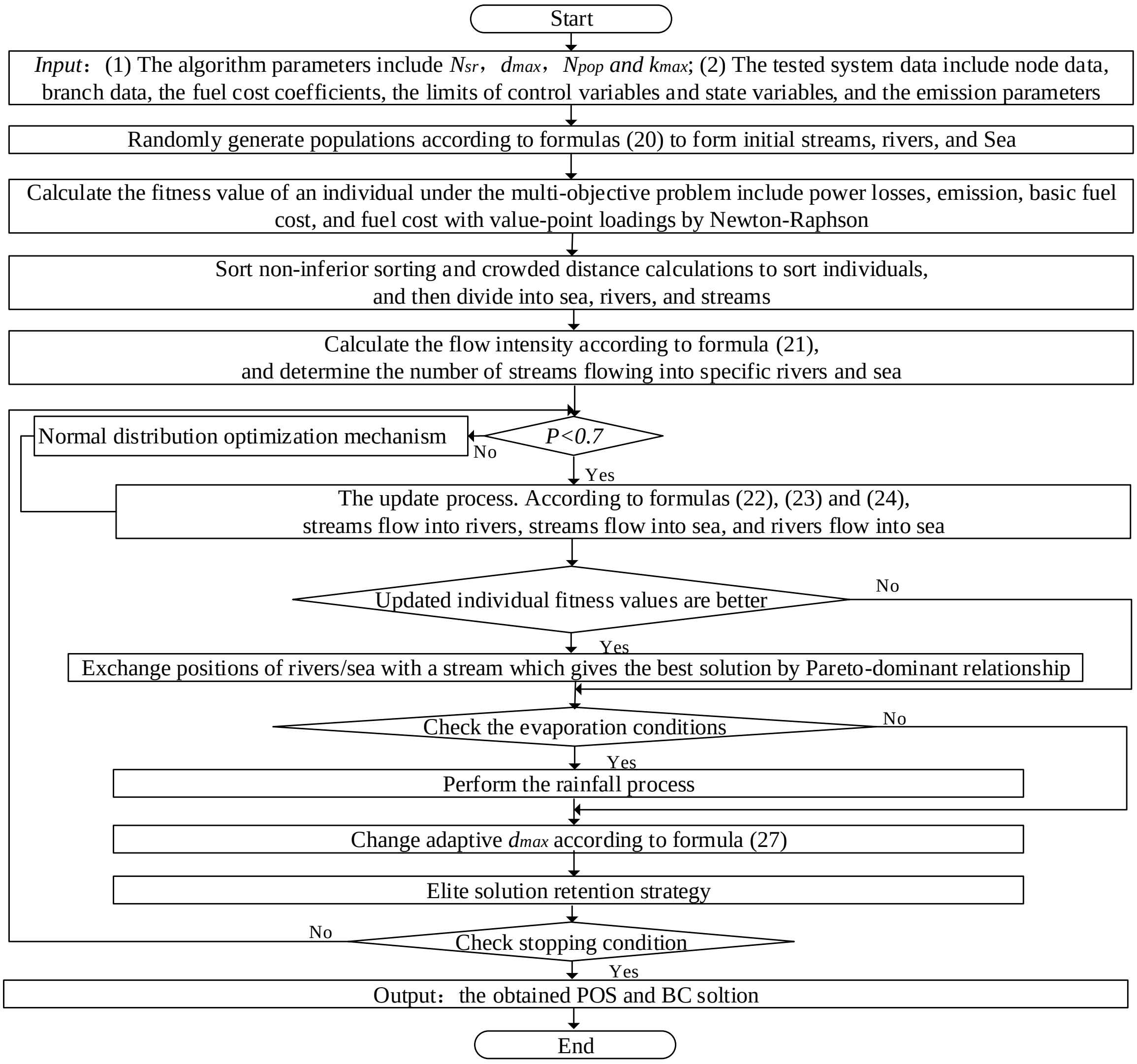

Fig. 1.The flow chart of MONIWCA to solve the MOOPF problem

TABLE III
THE SPECIFIC COMBINATION OF OBJECTIVE FUNCTIONS

The combination of objective functions	*Fcost*	*Ploss*	*Fcost-vp*	E_m	*L-index*	Test system
CASE 1	✔	✔				IEEE 30
CASE 2		✔	✔			IEEE 30
CASE 3		✔			✔	IEEE 30
CASE 4	✔	✔		✔		IEEE 30
CASE 5	✔			✔		IEEE 57
CASE 6	✔	✔				IEEE 57
CASE 7	✔	✔				IEEE 118

IV. Simulation and Result

The effectiveness of presenting the novel MOWCA algorithm can be validated by 7 optimal cases that are carried out on the MATLAB 2014a software and run on a PC with Intel(R) Core (TM) i5-4590 CPU @ 3.3GHZ with 8GB RAM, including 6 bi-objective and 1 tri-objectives. The cases have been examined and tested in IEEE 30, IEEE 57 and IEEE 118 system for solving MOOPF. Five objective functions are considered: *Ploss*, *Fcost*, *Em*, *Fcost-vp* and *L-index*.

A. Preparatory parameters

1) System parameters

The structure of the IEEE 30 system is shown in Fig. 2, including 6 generators, 4 transformers and 9 reactive power compensation devices. The system has a 24-dimensional vector. The transformer tap is within the range of 0.9-1.1 p.u and the voltage of generator buses and load buses are limited within the range of 0.95-1.1p.u. The generator coefficients of *Fcost* and *Em* in IEEE 30 system are shown in [22].

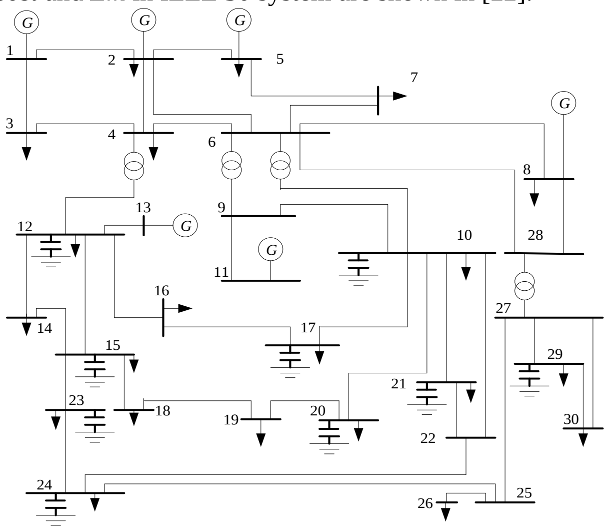

Fig. 2.The structure of the IEEE 30 system

The single line diagram of the IEEE 57 system is shown in Fig. 3, including 7 generators. Its detailed data are taken from [22]. The system has a 33-dimensional vector. The transformer taps are bounded in 0.9-1.1 p.u and the range of voltage magnitude for *PQ* and *PV* bus are restricted between 0.9 and 1.1 p.u. The shunt capacitor is limited within the range of 0-0.3 p.u.

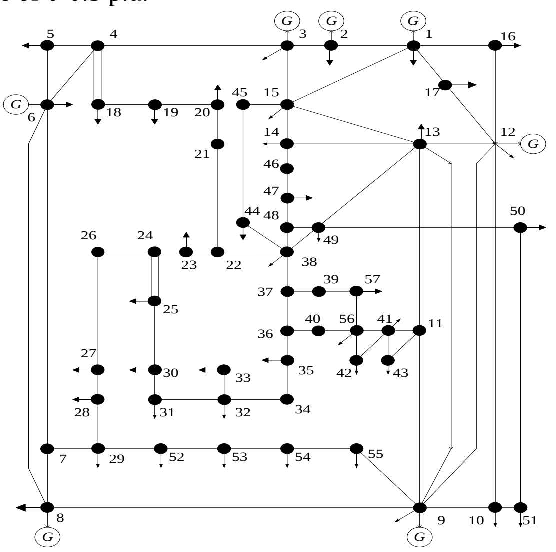

Fig. 3.The structure of the IEEE 57 system

The structure of the IEEE 118 system is shown in Fig. 4. The system has a 128-dimensional vector. The lower and upper limits of voltage magnitude for PV bus are 0.9-1.1 p.u. The limits of the transformer tap and limits of the shunt capacitor are consistent with the IEEE 57 system.

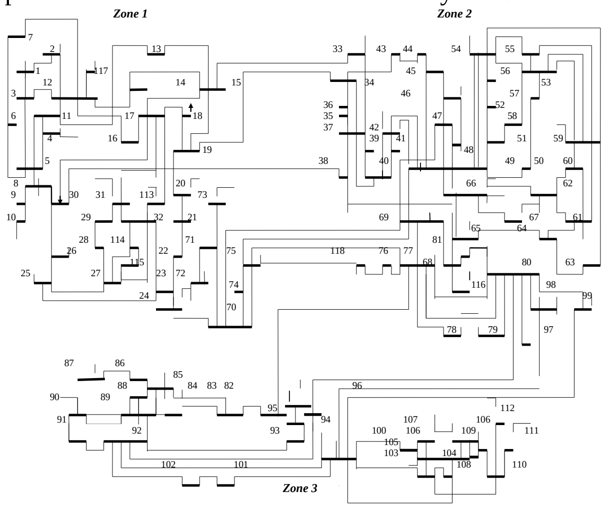

Fig. 4.The structure of the IEEE 118 system

2) Algorithm parameters

As shown in Fig. 5, *Fcost-vp* and *Ploss* are examples to conduct 100 to 500 dual-target simulation experiments. The Pareto curves obtained with different iteration times are also different. The Pareto front (PF) obtained in the 500 generation are closer to the real PFs. For the purpose of saving resources, the 400 generation and the 500 generation are infinitely close, and no more iterative experiments are needed. After repeated experiments, it was determined that the maximum number of iterations of the algorithm was the best for 500 generations. The parameter settings of the improved MONIWCA and MOPSO are shown in TABLE IV.

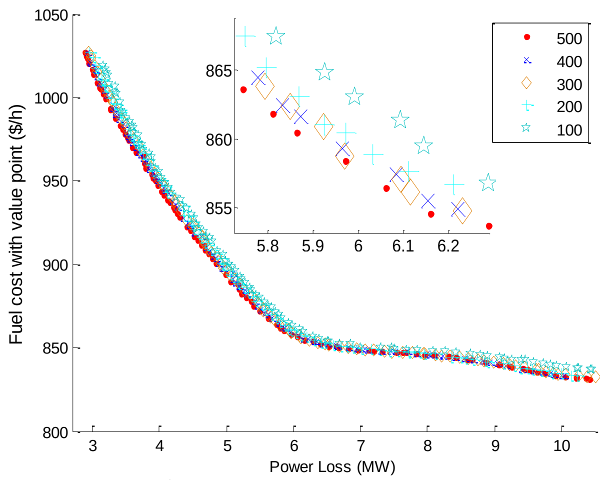

Fig. 5.The structure of the IEEE 118 system

TABLE IV
THE PARAMETER SETTINGS OF THE IMPROVED MOWCA AND MOPSO

Parameters	MONIWCA	Parameters	MOPSO
Population size: N_{pop}	100	Population size: N_{pop}	100
Max iterations: k_{max}	500	Max iterations: k_{max}	500
Rivers and sea: N_{Sr}	20	Learning factor: C_1	2
Update factor: C	2	Learning factor: C_2	2
Search range: μ	0.1	Weight coefficient: w	0.9

B. Trials on IEEE 30

1) CASE 1: Optimizing *Fcost* and *Ploss*

The Pareto[23] optimal solution (POS) obtained by the MOPSO and MONIWCA algorithms is shown in Fig. 6. POS is obtained based on the multi-objective strategy described in the previous section. As can be seen in Fig. 6, both the MOPSO and MONIWCA algorithms have obtained the PF, but MONIWCA has a more uniform and continuous Pareto solution set than MOPSO. Fig. 7 shows the minimum *Ploss*, the minimum *Fcost*, and the optimal POS obtained by the improved algorithm.

TABLE V indicates the control variable vectors. In this table, the BCs by the improved water cycle algorithm include 833.7570 $/h of *Fcost* and 5.0331 MW of *Ploss*. The optimal compromise solution obtained by MOPSO include 835.7867 $/h of *Fcost* and 5.2074 MW of *Ploss*. In order to make a more comprehensive comparison of the algorithms proposed in this paper, TABLE VI shows the BCs obtained by optimizing CASE 1 under the same conditions by consulting typical intelligent algorithms in recent years.

TABLE V
CONTROL VARIABLES OF BCS FOR CASE 1

control variables	MOPSO	MONIWCA
P_{G2}(MW)	60.9018	55.1847
P_{G5}(MW)	33.9575	33.2180
P_{G8}(MW)	33.549	34.4562
P_{G11}(MW)	23.7659	26.0294
P_{G13}(MW)	20.6372	22.2379
V_{G1}(p.u.)	1.0938	1.0992
V_{G2}(p.u.)	1.0850	1.0911
V_{G5}(p.u.)	1.0553	1.0677
V_{G8}(p.u.)	1.0724	1.0759
V_{G11}(p.u.)	1.0803	1.0808
V_{G13}(p.u.)	1.0889	1.0808
T_{11}(p.u.)	0.9759	1.0291
T_{12}(p.u.)	1.0141	0.9590
T_{15}(p.u.)	0.9751	1.0092
T_{36}(p.u.)	0.9910	0.9896
Q_{C10}(p.u.)	0.0360	0.0212
Q_{C12}(p.u.)	0.0261	0.0285
Q_{C15}(p.u.)	0.0494	0.0338
Q_{C17}(p.u.)	0.0458	0.0457
Q_{C20}(p.u.)	0.0000	0.0499
Q_{C21}(p.u.)	0.0186	0.0479
Q_{C23}(p.u.)	0.0368	0.0405
Q_{C24}(p.u.)	0.0490	0.0352
Q_{C29}(p.u.)	0.0044	0.0434
Obj1 ($/h)	835.7867	**833.7570**
Obj2 (MW)	5.2074	**5.0331**

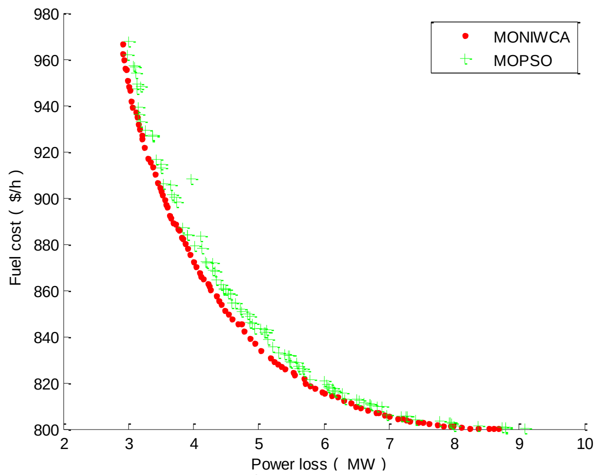

Fig. 6. Simulation PFs obtained for CASE 1

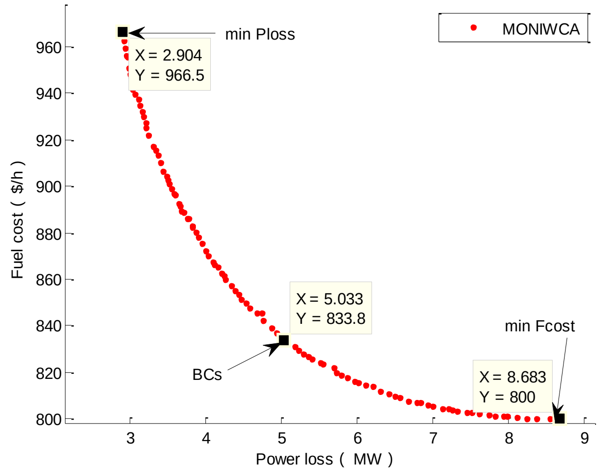

Fig. 7. Simulation PFs of MONIWCA obtained for CASE 1

TABLE VI
Comparison result of BCS for CASE 1

Algorithm	Fuel cost ($/h)	Ploss (MW)
NSGA-II [24]	837.4160	5.0397
MOTLBO [25]	830.7813	5.2742
MODFA [26]	833.9365	4.9561
MODE [27]	828.5900	5.6900

2) CASE 2: Optimizing *Fcost-vp* and *Ploss*

In practical application, the generator has the value-point effect and the fuel cost curve has non-differentiable points, therefore the optimization becomes a non-convex problem. Considering the valve point effect, in CASE 2, *Fcost-vp* and *Ploss* are minimized simultaneously.

We can observe that the proposed algorithms can obtain the Pareto front. The optimal compromise obtained by the two algorithms is shown in Fig. 8. It can be seen that the MONIWCA algorithm can better handle this dual-objective problem. Fig. 9 shows the minimum *Ploss*, the minimum *Fcost-vp*, and the optimal POS obtained by MONIWCA.

TABLE VII indicates the control variable vectors. In this table, the BCs by the improved water cycle algorithm includes 863.8368 $/h of *Fcost-vp* and 5.7637 MW of *Ploss*. The optimal compromise solution obtained by MOPSO includes 865.7202 $/h of *Fcost-vp* and 5.8294 MW of *Ploss*. The compromise solution obtained through MONIWCA algorithm is more advantageous than MOPSO algorithm. In order to make a more total comparison of the algorithms proposed in this paper, TABLE VIII shows the BCs obtained by optimizing CASE 2 under the same conditions by consulting typical intelligent algorithms in recent years.

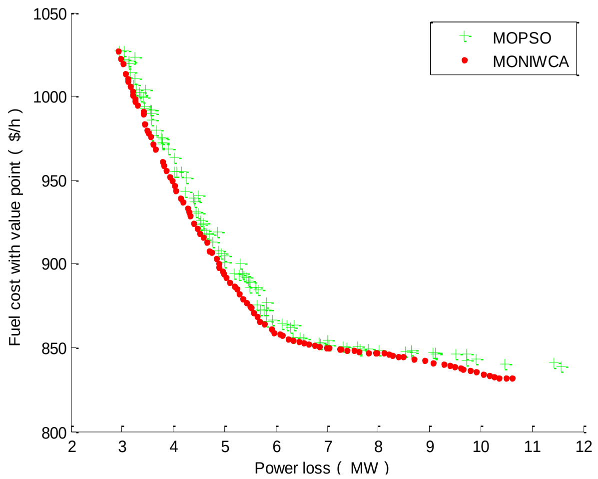

Fig. 8. Simulation PFs obtained for CASE 2

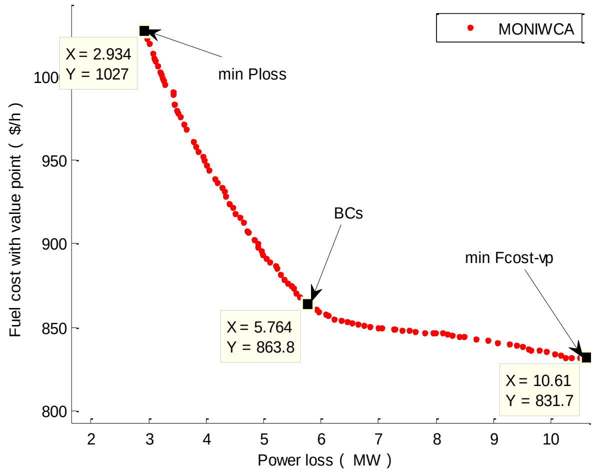

Fig. 9. Simulation PFs of MONIWCA obtained for CASE 2

TABLE VII
CONTROL VARIABLES OF BCS FOR CASE 2

control variables	MOPSO	MONIWCA
P_{G2}(MW)	43.0567	49.3536
P_{G5}(MW)	30.9781	30.0213
P_{G8}(MW)	35.0000	34.3831
P_{G11}(MW)	27.2070	28.4294
P_{G13}(MW)	16.7078	12.8268
V_{G1}(p.u.)	1.0981	1.0996
V_{G2}(p.u.)	1.0889	1.0879
V_{G5}(p.u.)	1.0572	1.0632
V_{G8}(p.u.)	1.0798	1.0756
V_{G11}(p.u.)	1.1000	1.0985
V_{G13}(p.u.)	1.0821	1.0927
T_{11}(p.u.)	1.0193	0.9608
T_{12}(p.u.)	0.9562	1.0997
T_{15}(p.u.)	0.9880	1.0529
T_{36}(p.u.)	1.0285	0.9973
Q_{C10}(p.u.)	0.0205	0.0238
Q_{C12}(p.u.)	0.0421	0.0382
Q_{C15}(p.u.)	0.0116	0.0482
Q_{C17}(p.u.)	0.0394	0.0248
Q_{C20}(p.u.)	0.0000	0.0455
Q_{C21}(p.u.)	0.0144	0.0499
Q_{C23}(p.u.)	0.0148	0.0480
Q_{C24}(p.u.)	0.0350	0.0471
Q_{C29}(p.u.)	0.0435	0.0489
Obj1 ($/h)	865.7202	**863.8368**
Obj2 (MW)	5.8294	**5.7637**

TABLE VIII
Comparison result of BCS for CASE 2

Algorithm	Fuel cost ($/h)	Ploss (MW)
NSGA-II [26]	871.5658	5.9469
NSGA-III [22]	865.9864	5.6847
MODFA [26]	857.7387	5.9252
NHBA [26]	868.9526	5.6761
DE [26]	865.9950	5.7665

3) CASE 3: Optimizing *L-index* and *Ploss*

The *L-index* and *Ploss* are optimized in CASE 3. We can observe that the proposed algorithms can obtain the Pareto front. Fig. 10 shows the optimal Pareto front distribution of MONIWCA and MOPSO in the IEEE 30 test system. As shown in Fig. 10, The uniformity and diversity of the PF of the MONIWCA algorithm is significantly better than the MOPSO algorithm. Fig. 11 shows the minimum *Ploss*, the minimum *L-index*, and the optimal POS obtained using the fuzzy satisfaction method obtained by the improved algorithm. TABLE IX indicates a comparison of optimal BCs. It shows that the MONIWCA methods can find better BCs.

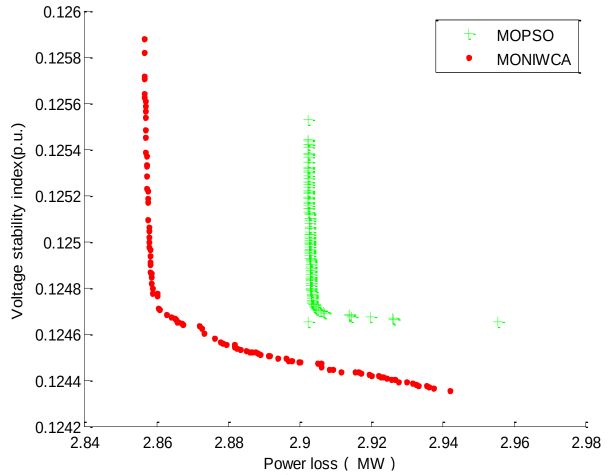

Fig. 10. Simulation PFs obtained for CASE 3

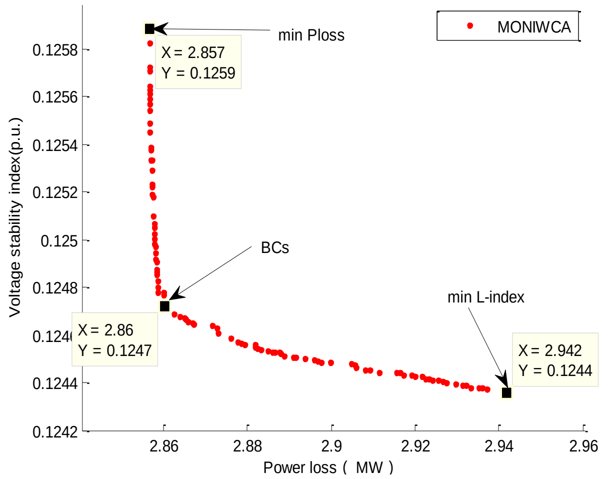

Fig. 11. Simulation PFs of MONIWCA obtained for CASE 3

TABLE IX
CONTROL VARIABLES OF BCS FOR CASE 3

control variables	MOPSO	MONIWCA
P_{G2}(MW)	80.0000	80.0000
P_{G5}(MW)	50.0000	50.0000
P_{G8}(MW)	35.0000	34.9988
P_{G11}(MW)	30.0000	30.0000
P_{G13}(MW)	40.0000	40.0000
V_{G1}(p.u.)	1.1000	1.1000
V_{G2}(p.u.)	1.1000	1.1000
V_{G5}(p.u.)	1.0866	1.0825
V_{G8}(p.u.)	1.1000	1.0910
V_{G11}(p.u.)	1.1000	1.0992
V_{G13}(p.u.)	1.1000	1.0998
T_{11}(p.u.)	1.0779	1.0455
T_{12}(p.u.)	0.9000	0.9000
T_{15}(p.u.)	0.9902	0.9893
T_{36}(p.u.)	0.9772	0.9704
Q_{C10}(p.u.)	0.0500	0.0382
Q_{C12}(p.u.)	0.0500	0.0484
Q_{C15}(p.u.)	0.0000	0.0241
Q_{C17}(p.u.)	0.0500	0.0000
Q_{C20}(p.u.)	0.0457	0.0500
Q_{C21}(p.u.)	0.0500	0.0500
Q_{C23}(p.u.)	0.0383	0.0500
Q_{C24}(p.u.)	0.0500	0.0500
Q_{C29}(p.u.)	0.0212	0.0214
Obj1(p.u)	0.1247	**0.1247**
Obj2(MW)	2.9038	**2.8609**

4) CASE 4: Optimizing *Fcost*, *Ploss* and *Em*

The *Fcost*, *Ploss* and *Em* are optimized in CASE 4. Fig. 12 shows the optimal Pareto front distribution of MONIWCA and MOPSO in the IEEE 30 test system. As shown in Fig. 12, The uniformity and diversity of the PF of the MONIWCA algorithm are significantly better than the MOPSO algorithm. Fig. 13 shows the minimum *Ploss*, the minimum *Fcost* and the optimal POS obtained by the improved algorithm. TABLE X indicates a comparison of optimal BCs. It intuitively states that the BCs obtained by MONIWCA algorithm include 879.9493 $/h of *Fcost*, 4.1744 MW of *Ploss* and 0.2171 ton/h of *Em*. The BCs obtained by MOPSO algorithm include 906.1371 $/h of *Fcost*, 4.5434 MW of *Ploss* and 0.2318 ton/h of *Em*. MONIWCA algorithm also has a huge advantage in dealing with trible objective problems.

TABLE X
CONTROL VARIABLES OF BCS FOR CASE 4

control variables	MOPSO	MONIWCA
P_{G2}(MW)	80.0000	60.2716
P_{G5}(MW)	50.0000	40.1422
P_{G8}(MW)	32.7228	35.0000
P_{G11}(MW)	20.1767	30.0000
P_{G13}(MW)	15.7144	33.0304
V_{G1}(p.u.)	1.1000	1.0986
V_{G2}(p.u.)	1.1000	1.0895
V_{G5}(p.u.)	1.1000	1.0634
V_{G8}(p.u.)	1.0811	1.0769
V_{G11}(p.u.)	1.0414	1.0978
V_{G13}(p.u.)	1.0218	1.1000
T_{11}(p.u.)	1.0281	1.0951
T_{12}(p.u.)	1.1000	0.9004
T_{15}(p.u.)	1.0026	1.0155
T_{36}(p.u.)	1.1000	0.9827
Q_{C10}(p.u.)	0.0500	0.0287
Q_{C12}(p.u.)	0.0500	0.0208
Q_{C15}(p.u.)	0.0107	0.0280
Q_{C17}(p.u.)	0.0489	0.0345
Q_{C20}(p.u.)	0.0500	0.0498
Q_{C21}(p.u.)	0.0500	0.0194
Q_{C23}(p.u.)	0.0400	0.0460
Q_{C24}(p.u.)	0.0000	0.0453
Q_{C29}(p.u.)	0.0500	0.0194
Obj1 ($/h)	0.2318	0.2171
Obj2 (ton/h)	906.1371	**879.9493**
Obj3 (MW)	4.5434	**3.8609**

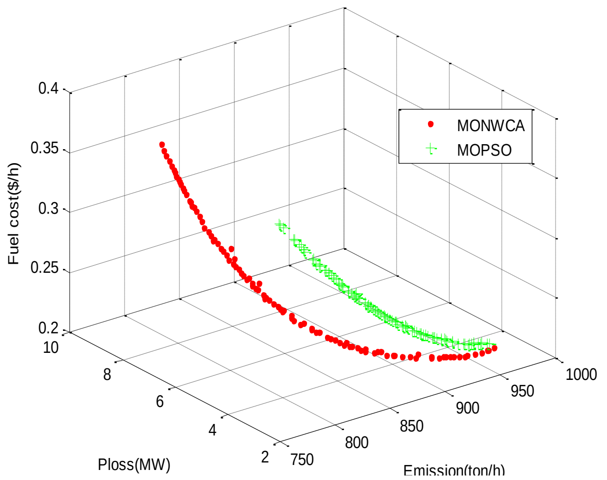

Fig. 12. Simulation PFs obtained for CASE 4

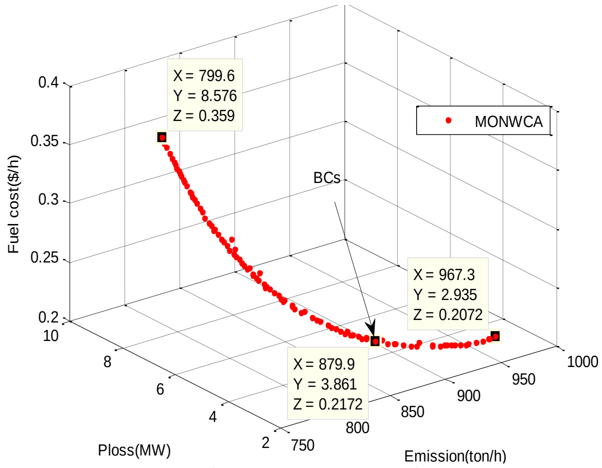

Fig. 13. Simulation PFs of MONIWCA obtained for CASE 4

C. Trials on IEEE 57

1) CASE 5: Optimizing *Em* and *Fcost*

In CASE 5, MONIWCA and MOPSO are tested for simultaneous minimization of *Em* and *Fcost* on IEEE 57. The results of simulation are presented in Fig. 14. Compared with the MOPSO algorithm, the PF obtained by the MONIWCA algorithm is clear and accurate. Fig. 15 shows the minimum *Fcost*, the minimum *Em*, and the optimal POS obtained by MONIWCA. TABLE XI indicates a comparison of optimal BCs. MONIWCA algorithm still exists competitiveness.

TABLE XI
CONTROL VARIABLES OF BCS FOR CASE 5

control variables	MOPSO	MONIWCA
P_{G2}(MW)	100.0000	99.8298
P_{G3}(MW)	92.5782	84.8746
P_{G6}(MW)	100.0000	99.9291
P_{G8}(MW)	341.1589	354.9838
P_{G9}(MW)	99.6328	99.9149
P_{G12}(MW)	343.1600	318.5327
V_{G1}(p.u.)	1.1000	1.0897
V_{G2}(p.u.)	1.1000	1.0822
V_{G3}(p.u.)	1.1000	1.0813
V_{G6}(p.u.)	1.1000	1.0968
V_{G8}(p.u.)	1.1000	1.0994
V_{G9}(p.u.)	1.1000	1.0940
V_{G12}(p.u.)	1.1000	1.0808
T_{19}(p.u.)	0.9584	1.0396
T_{20}(p.u.)	0.9920	1.0477
T_{31}(p.u.)	0.9712	1.0982
T_{35}(p.u.)	0.9959	1.0485
T_{36}(p.u.)	0.9769	1.0960
T_{37}(p.u.)	0.9754	1.0549
T_{41}(p.u.)	0.9493	1.0734
T_{46}(p.u.)	0.9546	1.0673
T_{54}(p.u.)	0.9000	0.9006
T_{58}(p.u.)	0.9560	0.9896
T_{59}(p.u.)	0.9797	0.9585
T_{65}(p.u.)	1.0000	1.0036
T_{66}(p.u.)	0.9992	0.9461
T_{71}(p.u.)	1.0000	1.0796
T_{73}(p.u.)	1.0000	1.0932
T_{76}(p.u.)	0.9344	1.0718
T_{80}(p.u.)	0.9625	1.0417
Q_{C18}(p.u.)	0.1141	0.1374
Q_{C25}(p.u.)	0.2739	0.2462
Q_{C53}(p.u.)	0.1446	0.2198
Obj1 ($/h)	42888.3500	**42808.7299**
Obj2 (ton/h)	1.3311	**1.3167**

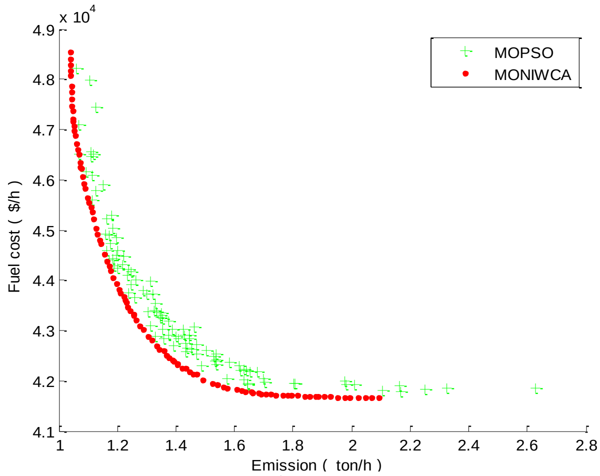

Fig. 14. Simulation PFs obtained for CASE 5

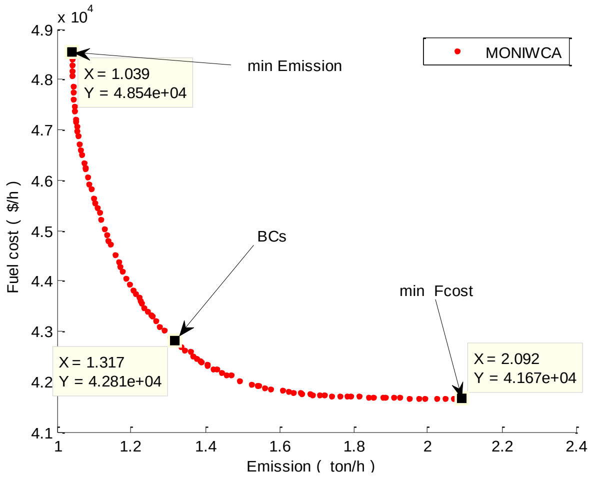

Fig. 15. Simulation PFs of MONIWCA obtained for CASE 5

2) CASE 6: Optimizing *Fcost* and *Ploss*

The achieved PF of the proposed MONIWCA algorithm, MOPSO algorithm is shown in Fig. 16. The *Fcost* and *Ploss* are optimized in CASE 6 on IEEE 57. It can be seen that the MONIWCA algorithm has great potential in achieving uniform distribution of PF. Fig. 17 shows the minimum *Ploss*, the minimum *Fcost*, and the optimal POS obtained using the fuzzy satisfaction method obtained by MONIWCA.

TABLE XII indicates a comparison of BCs. In this table, the BCs by MONIWCA include 42094.4600 $/h of *Fcost* and 10.3153 MW of *Ploss*. The BCs obtained by MOPSO include 42088.3800 $/h of *Fcost* and 11.7191 MW of *Ploss*. The MOPSO algorithm is slightly better than the MONIWCA algorithm in terms of saving fuel costs, but the MONIWCA algorithm can greatly reduce *Ploss*.

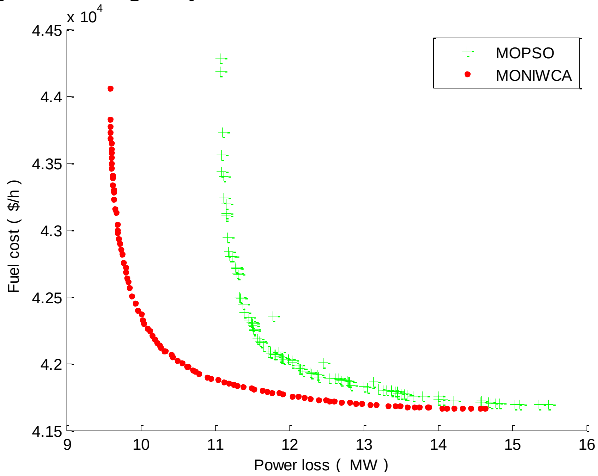

Fig. 16. Simulation PFs obtained for CASE 6

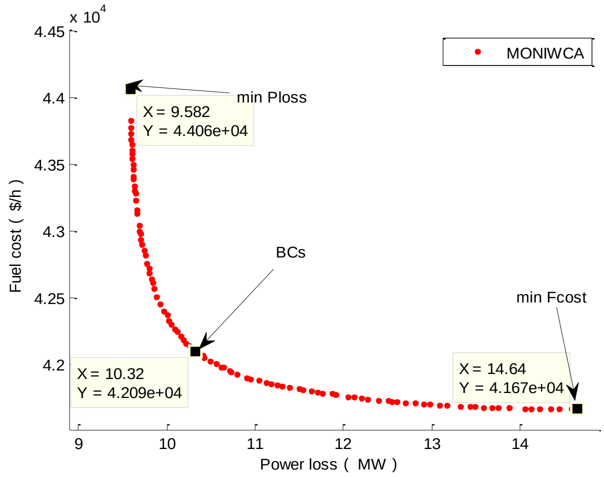

Fig. 17. Simulation PFs of MONIWCA obtained for CASE 6

TABLE XII
CONTROL VARIABLES OF BCS FOR CASE 6

control variables	MOPSO	MONIWCA
P_{G2}(MW)	100.0000	72.7378
P_{G3}(MW)	55.4432	60.5458
P_{G6}(MW)	100.0000	98.4866
P_{G8}(MW)	358.3027	359.0951
P_{G9}(MW)	100.0000	99.6130
P_{G12}(MW)	410.0000	409.8796
V_{G1}(p.u.)	1.1000	1.0894
V_{G2}(p.u.)	1.1000	1.0869
V_{G3}(p.u.)	1.1000	1.0846
V_{G6}(p.u.)	1.1000	1.0933
V_{G8}(p.u.)	1.1000	1.0965
V_{G9}(p.u.)	1.1000	1.0826
V_{G12}(p.u.)	1.1000	1.0718
T_{19}(p.u.)	0.9938	0.9648
T_{20}(p.u.)	1.1000	1.0955
T_{31}(p.u.)	0.9855	1.0179
T_{35}(p.u.)	1.1000	0.9334
T_{36}(p.u.)	1.1000	1.0789
T_{37}(p.u.)	1.0453	0.9928
T_{41}(p.u.)	1.1000	0.9925
T_{46}(p.u.)	0.9857	0.9399
T_{54}(p.u.)	0.9000	0.9001
T_{58}(p.u.)	1.1000	0.9662
T_{59}(p.u.)	1.0413	0.9796
T_{65}(p.u.)	1.1000	0.9603
T_{66}(p.u.)	1.0032	0.9462
T_{71}(p.u.)	1.0194	0.9691
T_{73}(p.u.)	0.9000	1.024
T_{76}(p.u.)	0.9614	0.9734
T_{80}(p.u.)	1.1000	0.9991
Q_{C18}(p.u.)	0.0526	0.1643
Q_{C25}(p.u.)	0.2801	0.1280
Q_{C53}(p.u.)	0.1709	0.1362
Obj1 ($/h)	42088.3800	**42094.4600**
Obj2(ton/h)	11.7191	**10.3153**

D. Trials on IEEE 118

1) CASE 7: Optimizing *Fcost and Ploss*

In CASE 7, the proposed algorithm and the MOPSO are tested for the simultaneous minimization of *Fcost* and *Ploss* on IEEE 118. However, The MOPSO algorithm did not find a uniform Pareto [28] solution set within 500 generations and could not draw an effective curve. Fig. 18 shows only the PF of MONIWCA. This also shows that MONIWCA has a strong ability to optimize the super large power system, while the MOPSO algorithm is a bit different. Additionally, TABLE XIII shows the control variables and BCs. As is shown in TABLE XIII, by comparing the references in [22], we find the BCs and control variables of the NSGA-III in the same system. By comparing the published articles, it has more comparative value.

In detail, the BCs obtained by MONIWCA algorithm include 58258 \$/h of *Fcost* and 49.7308 MW of *Ploss*. The BCs obtained by NSGA-III include 59474.4030 \$/h of *Fcost* and 58.4603 MW of *Ploss*. By comparing the optimal compromise solutions of the two algorithms, the two objective function values of the optimal compromise solution obtained by MONIWCA are both less than NSGA-III. On large-scale systems, MONIWCA also shows a good ability to find the optimal solution.

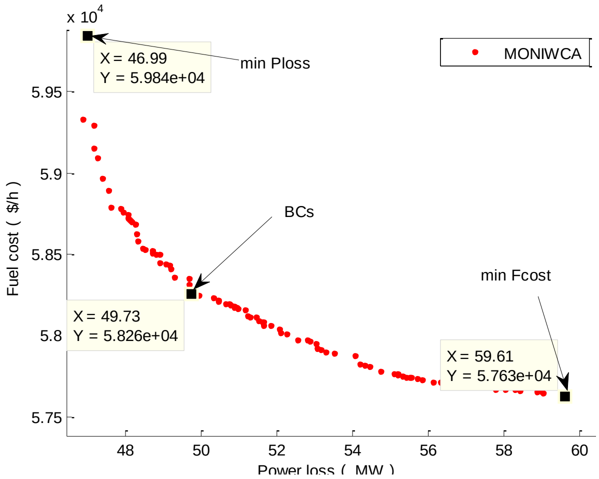

Fig. 18. Simulation PFs of MONIWCA obtained for CASE 7

E. Evaluation Index

Performance indicators are used to evaluate whether the algorithm achieves the desired goals[29]. The above simulation only reflect the optimal compromise solution among the 30 times obtained by the two algorithms. In order to more comprehensively evaluate the effectiveness of the improved algorithm, two indicators in multi-objective problem, the GD[30] and the SP [31] are used for statistical analysis of 30 results. The computation complexity is introduced to quantitatively assess the efficiency of the algorithm.

1) GD

The GD index is often used to measure the convergence of algorithms in multi-objective problems. The definition of GD index is shown in formula (36). It calculates the distance between the PF solution set and the real PF obtained by the intelligent algorithm, which can be used to measure the convergence of the PF solution set. Generally, the smaller the GD index value means the better the consistency and convergence between the obtained PF frontier and the reference PF. The GD index boxplots for simulation CASE 1 to CASE 6 are shown in Fig. 19.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} de_i^2}}{n} \tag{36}$$

2) SP

The SP indicator measures the standard deviation of the minimum distance of each solution to other solutions. The definition of SP index is shown in formula (37). The SP indicator is usually used to measure the uniformity of the POS solution set. Generally, the smaller the *SP* index value of the solution set, the more uniform the distribution of the solution set, and the greater the competitive advantage of the solution obtained. Fig. 20 shows the *SP* index box diagrams used to simulate CASE 1 to CASE 6.

$$Spacing(P) = \sqrt{\frac{1}{|P|-1}\sum_{I=1}^{|P|}(d_a - d_i)^2} \tag{37}$$

where represents the minimum distance from the d_i solution to other solutions in P, and d_a represents the mean of all d_i.

3) Analysis of evaluation index results

Boxplot can visually display data distribution characteristics. The advantages and disadvantages of the two algorithms can be distinguished by comparing the median, outliers and distribution interval of the boxplot.

Fig. 19 shows the GD indicators of CASE 1 to CASE 6. Since the experiment CASE 7 MOPSO algorithm does not obtain valid results, it does not give boxplot. It can be seen from Fig. 19 that the MONIWCA algorithm is very evenly distributed and has few outliers. The mean is less than the MOPSO algorithm. The results show that the PF obtained by MONIWCA is closer to the actual situation and has strong convergence. Fig. 20 shows the SP indicators of the two algorithms. As shown in Fig. 20, The MONIWCA algorithm is generally well distributed and has a strong competitive advantage, but it is slightly inferior to MOPSO in CASE 4, CASE 5.

TABLE XIV calculates the average and deviation of the two indicators, GD and SP. Observation from the specific calculation value also verifies the conclusions drawn in Fig. 19 and Fig.20. The MONIWCA algorithm can obtain better BCs, and its convergence, extensiveness and stability also surpass the MOPSO algorithm.

F. Computation complexity

The computation complexity is one of the common evaluation indexes, which can be represented by running time, to measure the performance of modified algorithms[32]. An efficient algorithm should shorten the search time as much as possible without affecting the optimization quality. TABLE XV states the average running time of the two algorithms running 7 simulation cases respectively. Under the same number of iterations, the running time of MONIWCA in experiment CAES1 is 299.0500 seconds, and the running time of MOPSO is 367.8034 seconds. In experiment CASE2, the running time of MONIWCA is 287.3517 seconds, and the running time of MOPSO is 372.2897 seconds. In Experimental CASE 3, the running time of MONIWCA is 264.6665 seconds, while the running time of MOPSO is 380.272 seconds. In case the experiment 4, MONIWCA running time of 387.1648 seconds, while the running time MOPSO is 404.3803 seconds. In the test CASE 5, the runtime MONIWCA is 413.4143 seconds, while the running time MOPSO is 472.1514 seconds. In CASE 6, the running time of MONIWCA is 607.717 seconds, and the running time of MOPSO is 472.1514 seconds. The running time efficiency of MONIWCA algorithm is much higher than MOPSO. In the experiment CASE7, because MOPSO did not run a valid experiment result, the running time was not recorded. It can be seen from the TABLE XV that MONIWCA algorithm iterates 500 times in a shorter time than MOPSO algorithm, and it is more efficient to find the optimal solution, which has great advantages.

TABLE XIII
CONTROL VARIABLES OF BCS FOR CASE 7

control variables	MONIWCA	NSGA-III[22]	control variables	MONIWCA	NSGA-III[22]	control variables	MONIWCA	NSGA-III[22]
P_{G4}(MW)	5.1407	5.0000	P_{G100}(MW)	109.4305	100.1413	V_{G74}(p.u.)	0.9794	1.0164
P_{G6}(MW)	5.0861	22.7851	P_{G103}(MW)	8.4821	8.2474	V_{G76}(p.u.)	0.9962	1.0271
P_{G8}(MW)	5.7205	7.2779	P_{G104}(MW)	26.6807	38.9282	V_{G77}(p.u.)	1.0161	1.0354
P_{G10}(MW)	182.6153	186.5229	P_{G105}(MW)	35.8219	41.3443	V_{G80}(p.u.)	1.0000	0.9943
P_{G12}(MW)	208.9950	234.1053	P_{G107}(MW)	11.4164	8.6289	V_{G85}(p.u.)	1.0053	0.9887
P_{G15}(MW)	11.2950	12.5793	P_{G110}(MW)	25.0019	26.5478	V_{G87}(p.u.)	0.9345	0.9780
P_{G18}(MW)	73.5972	46.8689	P_{G111}(MW)	26.7671	25.3709	V_{G89}(p.u.)	1.0245	1.0065
P_{G19}(MW)	5.5902	21.1907	P_{G112}(MW)	26.3646	33.4090	V_{G90}(p.u.)	0.9990	1.005
P_{G24}(MW)	5.0160	8.6509	P_{G113}(MW)	43.3264	27.5689	V_{G91}(p.u.)	1.0177	0.9979
P_{G25}(MW)	112.4425	127.7909	P_{G116}(MW)	26.3556	25.1972	V_{G92}(p.u.)	1.0237	1.0104
P_{G26}(MW)	223.2441	218.8630	V_{G1}(p.u.)	1.0108	1.0083	V_{G99}(p.u.)	1.0387	0.9772
P_{G27}(MW)	9.2709	12.9486	V_{G4}(p.u.)	1.0046	1.0027	V_{G100}(p.u.)	1.0370	1.0034
P_{G31}(MW)	19.4343	21.4135	V_{G6}(p.u.)	1.0079	0.9955	V_{G103}(p.u.)	1.0153	1.0035
P_{G32}(MW)	76.2148	50.5916	V_{G8}(p.u.)	0.9882	1.0364	V_{G104}(p.u.)	1.0066	0.9992
P_{G34}(MW)	8.1083	8.4393	V_{G10}(p.u.)	1.0079	0.9974	V_{G105}(p.u.)	1.0040	0.9919
P_{G36}(MW)	91.2198	69.4697	V_{G12}(p.u.)	1.0155	1.0062	V_{G107}(p.u.)	1.0059	1.0006
P_{G40}(MW)	8.6951	9.0326	V_{G15}(p.u.)	1.0154	1.0219	V_{G110}(p.u.)	0.9785	0.9753
P_{G42}(MW)	8.2109	21.9630	V_{G18}(p.u.)	1.0137	1.0169	V_{G111}(p.u.)	1.0096	0.9486
P_{G46}(MW)	33.0844	53.5697	V_{G19}(p.u.)	1.0033	1.0115	V_{G112}(p.u.)	0.9689	0.9800
P_{G49}(MW)	249.9981	164.9918	V_{G24}(p.u.)	1.0193	1.0285	V_{G113}(p.u.)	1.0350	0.9911
P_{G54}(MW)	246.0307	216.8517	V_{G25}(p.u.)	1.0422	1.0501	V_{G116}(p.u.)	0.9929	1.0136
P_{G55}(MW)	28.0063	56.3268	V_{G26}(p.u.)	1.0040	0.9895	T_{8}(p.u.)	0.9856	0.9698
P_{G56}(MW)	28.7037	81.5120	V_{G27}(p.u.)	1.0096	0.9609	T_{32}(p.u.)	1.0219	0.9407
P_{G59}(MW)	197.7345	121.4729	V_{G31}(p.u.)	1.01288	0.9647	T_{36}(p.u.)	1.0242	1.0444
P_{G61}(MW)	99.8390	199.0160	V_{G32}(p.u.)	1.0115	1.0110	T_{51}(p.u.)	1.0102	0.9569
P_{G62}(MW)	56.1174	26.7623	V_{G34}(p.u.)	1.0128	1.0121	T_{93}(p.u.)	1.0015	0.9420
P_{G65}(MW)	275.2493	258.6241	V_{G36}(p.u.)	1.0056	1.0005	T_{95}(p.u.)	1.1000	0.9289
P_{G66}(MW)	239.4934	201.4848	V_{G40}(p.u.)	1.0163	1.0029	T_{102}(p.u.)	0.9875	1.0886
P_{G69}(MW)	41.6746	57.7153	V_{G42}(p.u.)	1.0317	1.0358	T_{107}(p.u.)	0.9996	0.9398
P_{G70}(MW)	10.0090	10.8531	V_{G46}(p.u.)	1.0254	1.0185	T_{127}(p.u.)	1.0054	1.0012
P_{G72}(MW)	5.4660	13.8617	V_{G49}(p.u.)	1.0092	1.0061	Q_{C34}(p.u.)	0.0014	0.1113
P_{G73}(MW)	5.0494	6.2214	V_{G54}(p.u.)	1.0045	1.0029	Q_{C44}(p.u.)	0.0662	0.0250
P_{G74}(MW)	58.0688	30.6756	V_{G55}(p.u.)	1.0062	1.0008	Q_{C45}(p.u.)	0.0171	0.1463
P_{G76}(MW)	67.3902	65.7396	V_{G56}(p.u.)	1.0044	1.0195	Q_{C46}(p.u.)	0.0199	0.2550
P_{G77}(MW)	150.1652	160.2489	V_{G59}(p.u.)	1.0008	1.0258	Q_{C48}(p.u.)	0.1714	0.0055
P_{G80}(MW)	53.8895	27.6238	V_{G61}(p.u.)	0.9975	1.0104	Q_{C74}(p.u.)	0.0000	0.2235
P_{G85}(MW)	10.2452	15.1266	V_{G62}(p.u.)	1.0025	1.0289	Q_{C79}(p.u.)	0.1492	0.2282
P_{G87}(MW)	129.7834	157.8482	V_{G65}(p.u.)	1.0297	1.0118	Q_{C82}(p.u.)	0.1379	0.0616
P_{G89}(MW)	51.0747	67.3263	V_{G66}(p.u.)	1.0133	1.0595	Q_{C83}(p.u.)	0.1610	0.1659
P_{G90}(MW)	8.0033	9.5007	V_{G69}(p.u.)	1.0170	1.0453	Q_{C105}(p.u.)	0.1365	0.2435
P_{G91}(MW)	28.6133	28.7783	V_{G70}(p.u.)	0.9895	0.9785	Q_{C107}(p.u.)	0.2798	0.2120
P_{G92}(MW)	121.3511	102.4969	V_{G72}(p.u.)	0.9909	1.0294	Q_{C110}(p.u.)	0.1167	0.1663
P_{G99}(MW)	100.3416	101.0885	V_{G73}(p.u.)	1.0010	1.0504	Obj1 ($/h)	**58258.0000**	59474.4030
						Obj2(MW)	**49.7308**	58.4603

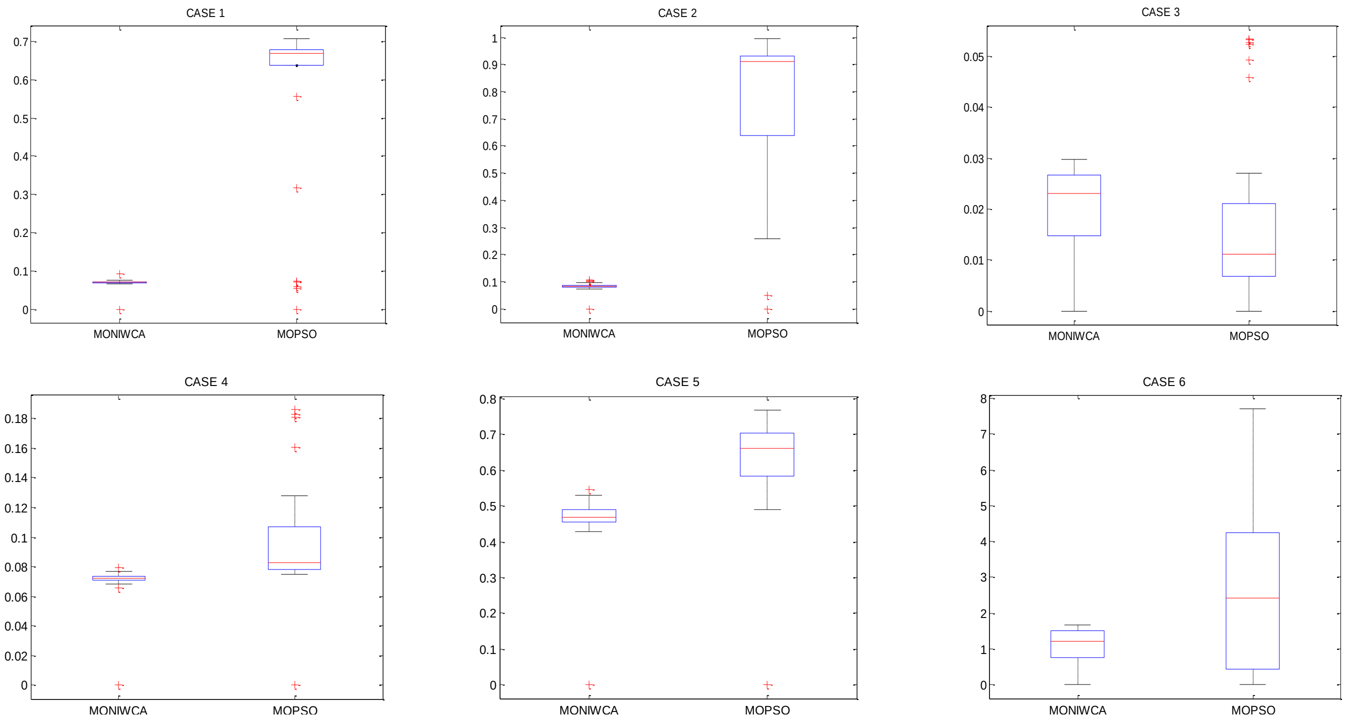

Fig. 19. Boxplots of GD from CASE 1 to CASE 6.

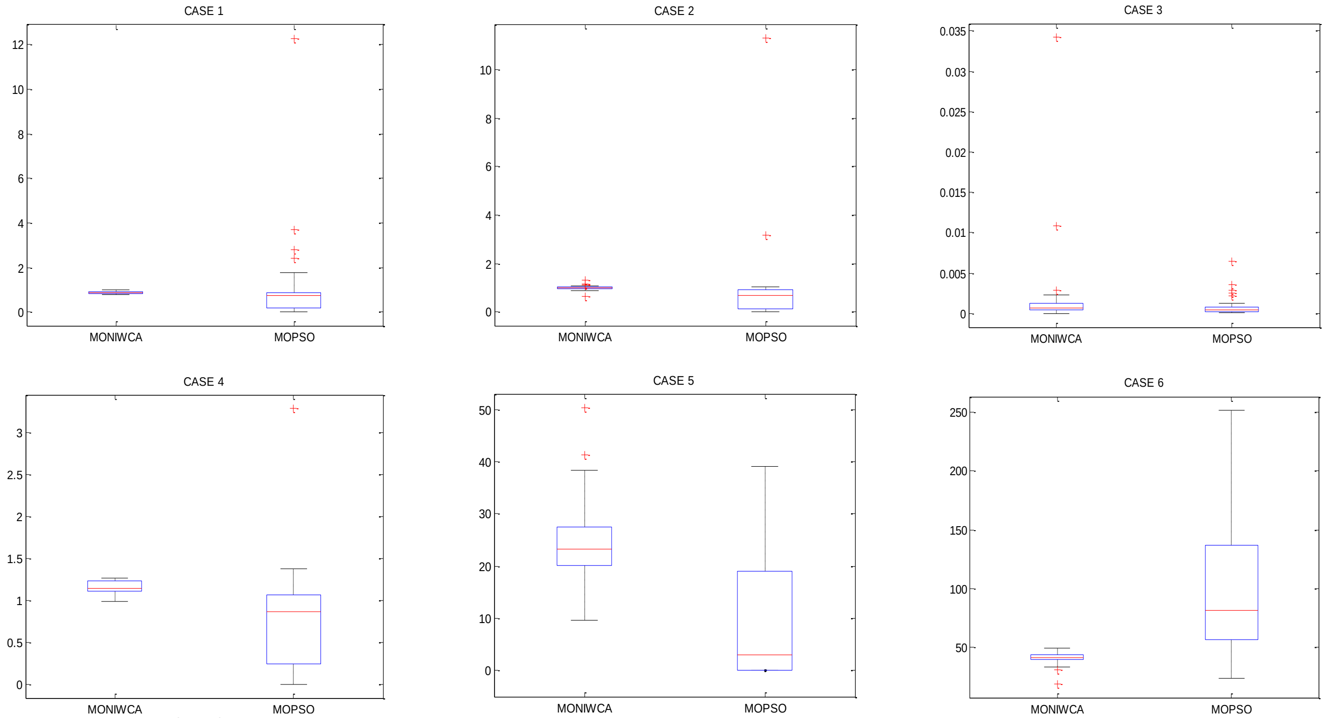

Fig. 20. Boxplots of SP from CASE 1 to CASE 6.

TABLE XIV
THE EVALUATION INDEX RESULTS OF THE TWO ALGORITHMS

index	GD				SP			
algorithm	MONIWCA		MOPSO		MONIWCA		MOPSO	
	average	deviation	average	deviation	average	deviation	average	deviation
CASE 1	0.0687	0.0138	0.5525	0.2381	0.8664	0.0617	1.2091	2.2533
CASE 2	0.0819	0.0177	0.7441	0.2943	0.9968	0.1003	0.9899	2.0399
CASE 3	0.0207	0.0080	0.0184	0.0176	0.0022	0.0063	0.0013	0.0010
CASE 4	0.0699	0.0134	0.0980	0.0383	1.1528	0.0801	0.7728	0.6550
CASE 5	1.0898	0.4427	2.6889	2.4395	24.1805	9.0487	10.3708	12.3802
CASE 6	0.4574	0.0906	0.6261	0.1366	41.1462	5.6392	100.8049	61.4656
CASE 7	0.6688	0.2874	-	-	19.3839	12.4494	-	-

TABLE XV
AVERAGE RUNNING TIME

Algorithm	CASE 1	CASE 2	CASE 3	CASE 4	CASE 5	CASE 6	CASE 7
MONIWCA	299.0500	287.3517	264.6665	387.1648	413.4143	607.7177	1674.2870
MOPSO	367.8034	372.2797	380.272	404.3803	472.1514	609.1775	-

V. CONCLUSION

In this paper, multi-objectives, such as the fuel cost, the fuel cost with value-point loadings, the emission, the voltage stability index and the power losses are considered to constitute different OPF problems with complex constraints. In view of the standard water cycle algorithm is difficult to solve the multi-objective optimization problem, this paper proposes a new multi-objective water cycle algorithm including the introduction of an evaporation process and normal distribution optimization mechanism to solve the MOOPF problem. MONIWCA is successfully applied to IEEE 30, IEEE 57 and IEEE 118 standard test systems including 7 test cases. The modified MOWCA also employs a constraint dominant strategy to guarantee zero constraint violations.

The obtained results confirm that MONIWCA can provide a more uniform and continuous Pareto solution and an advantageous compromise solution than MOPSO. Statistical analysis of SP and GD indicators proves that MONIWCA has a competitive advantage in the combined case and the algorithm has high convergence and strong stability. Therefore, it can be reasonably explained that the MONIWCA algorithm has a certain reference value for solving multi-objective optimization problems.

REFERENCES

[1] W. Warid, H. Hizam, N. Mariun and N.I.A. Wahab, "A novel quasi-oppositional modified Jaya algorithm for multi-objective optimal power flow solution," *Applied Soft Computing*, vol. 65, pp. 360-373, 2018.

[2] C.B. Teodoro, C.M. Viviana and D.S.C. Leandro, "Multi-objective optimization of the environmental-economic dispatch with reinforcement learning based on non-dominated sorting genetic algorithm," *Applied Thermal Engineering: Design, Processes, Equipment, Economics*, vol. 146, pp. 688-700, 2019.

[3] J. Xiong, C. Liu, Y. Chen, and S. Zhang, "A non-linear geophysical inversion algorithm for the mt data based on improved differential evolution", *Engineering Letters*, vol. 26, no. 1, pp. 161-170, 2018.

[4] X. Yuan, B. Zhang, and P. Wang, "Multi-objective optimal power

flow based on improved strength Pareto evolutionary algorithm," *Energy*, vol. 122, pp. 70-82, 2017
[5] L. Sun, J. Hu and H. Chen, "Artificial Bee Colony Algorithm Based on K-Means Clustering for Multi-objective Optimal Power Flow Problem," *Mathematical Problems in Engineering: Theory, Methods and Applications*, vol. 2015, no. 7, pp. 762851-762853, 2015.
[6] S. A. El-Sattar, S. Kamel, R.A. EI-Sehiemy, F. Jurado and J. Yu, "Single and multi-objective optimal power flow frameworks using Jaya optimization technique," *Neural Computing & Applications*, vol. 31, no. 12, pp. 8787-8806, 2019.
[7] F. Zou, D. Chen and Q. Xu, "A Survey of Application and Classification on Teaching-Learning-Based Optimization Algorithm," *Neurocomputing*, vol. 335, no. 5, pp. 366-383, 2019.
[8] B. Jeddi, A. H. Einaddin and R. Kazemzadeh, "A novel multi‐objective approach based on improved electromagnetism‐like algorithm to solve optimal power flow problem considering the detailed model of thermal generators," *International Transactions on Electrical Energy Systems*, vol. 27, no. 4, pp. 2293-2299, 2017.
[9] Y. Yuan, X. Wu, P. Wang and X. Yuan, "Application of improved bat algorithm in optimal power flow problem," *Applied Intelligence*, vol. 48, no. 8, pp. 2304-2314, 2018.
[10] J. Zhang, Q. Tang, P. Li, D. Deng and Y. Chen, "A modified MOEA/D approach to the solution of multi-objective optimal power flow problem," *Applied Soft Computing*, vol. 47, no. 10, pp. 494-514, 2016.
[11] G. Chen, L. Liu, P. Song and Y. Du, "Chaotic improved PSO-based multi-objective optimization for minimization of power losses and L index in power systems," *Energy conversion & Management*, vol. 86, no. 10, pp. 548-560, 2014.
[12] H. Eskandar, A. Sadollah, A. Bahreininejad and M. Hamdi, "Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Computers & Structures*, vol. 110-111, no.11, pp. 151-166, 2012.
[13] A.A. Heidari, R.A. Abbaspour and A.R. Jordehi, "An efficient chaotic water cycle algorithm for optimization tasks," *Neural computing & Applications*, vol. 28, no. 1, pp. 57-85, 2017.
[14] A. Korashy, S. Kamel, A. Youssef, and F. Jurado, "Modified water cycle algorithm for optimal direction overcurrent relays coordination," *Applied Soft Computing*, vol. 74, no. 1, pp. 10-25, 2019.
[15] K. Qaderi, S. Akbarifard, M.R. Madadi and B. Bakhtiari, "Optimal operation of multi-reservoirs by water cycle algorithm," *Proceedings of the Institution of Civil Engineers*, vol. 171, no. 4, pp. 179-190, 2018.
[16] E.A. Elhay and M.M. Elkholy, "Optimal dynamic and steady‐state performance of switched reluctance motor using water cycle algorithm," *IEEE Transactions on Electrical and Electronic Engineering*, vol. 13, no. 6, pp. 882-890, 2018.
[17] A. Sadollah, H. Eskandar, A. Bahreininejad and J.H. Kim, "Water cycle algorithm with evaporation rate for solving constrained and unconstrained optimization problems," *Applied Soft Computing*, vol. 30, no. 5, pp. 58-71, 2015.
[18] O. Bozorg-Haddad, M. Moravej and H.A. Loaiciga, "Application of the Water Cycle Algorithm to the Optimal Operation of Reservoir Systems," *Journal of Irrigation and Drainage Engineering-Asce*, vol. 141, no. 5, pp. 401-416, 2015.
[19] K. Guney and S. Basbug, "A quantized water cycle optimization algorithm for antenna array synthesis by using digital phase shifters," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 25, no. 1, pp. 21-29, 2015.
[20] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol.6, no. 2, pp. 182-197, 2002.
[21] A.A. Heidari, R.A. Abbaspour and A.R. Jordehi, "Gaussian bare-bones water cycle algorithm for optimal reactive power dispatch in electrical power systems," *Applied Soft Computing*, vol. 57, pp. 657-671, 2017.
[22] G. Chen, J. Qian, Z. Zhang and Z. Sun, "Applications of Novel Hybrid Bat Algorithm with Constrained Pareto Fuzzy Dominant Rule on Multi-Objective Optimal Power Flow Problems," *IEEE Access*, vol. 7, pp. 52060-52084, 2019.
[23] A. Formisano, M. Cioffi, R. Martone. "Pareto swarm optimisation of high temperature superconducting generators", *International Journal of Applied Electromagnetics & Mechanics*, vol. 15, no. 1., pp. 273-279, 2007.
[24] S.S. Reddy and C.S. Rathnam, "Optimal Power Flow using Glowworm Swarm Optimization," *International Journal of Electrical Power and Energy Systems*, vol. 80, no. 9., pp. 128-139, 2016.
[25] R.V. Rao, D.P. Rai and J. Balic, "Multi-objective optimization of machining and micro-machining processes using non-dominated sorting teaching-learning-based optimization algorithm," *Journal of Intelligent Manufacturing*, vol. 29, no. 8, pp. 1715-1737, 2018.
[26] G. Chen, X. Yi, Z. Zhang and H. Wang, "Applications of multi-objective dimension-based firefly algorithm to optimize the power losses, emission, and cost in power systems," *Applied Soft Computing*, vol. 68, no. 7, pp. 322-342, 2018.
[27] K. Rajalashmi and S.U. Prabha, "Hybrid Swarm Algorithm for Multi-objective Optimal Power Flow Problem," *Circuits and Systems*, vol. 7, no. 11, pp. 3589-3603, 2016.
[28] C.A.C. Coello, G.T. Pulido, and M.S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, pp. 256-279, 2004.
[29] G. Chen, J. Cao, Z. Zhang, and Z. Sun, "Application of Imperialist Competitive Algorithm with Its Enhanced Approaches for Multi-Objective Optimal Reactive Power Dispatch Problem," *Engineering Letters*, vol. 27, no. 8, pp. 579-592, 2019.
[30] M. Gertz and J. Krieter, "About the classification of abattoir data by using the boxplot-methodology and its derivatives", *Zuchtungskunde*, vol. 89, no. 6, pp. 401-412, 2017.
[31] L. While, P. Hingston, L. Barone and S. Huband, "A Faster Algorithm for Calculating Hypervolume," *IEEE Transations on Evolutionary Computation*, vol. 10, no. 1, pp. 29-38, 2006.
[32] G. Chen, J. Qian, Z. Zhang, and Z. Sun, "Many-objective New Bat Algorithm and Constraint-Priority Non-Inferior Sorting Strategy for Optimal Power Flow," *Engineering Letters*, vol. 27, no. 11, pp. 882-892, 2019.