Journal of Network Intelligence ©2021 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **6**, Number **2**, May **2021**

Advanced Equilibrium Optimizer for Electric Vehicle Routing Problem with Time Windows

Jeng-Shyang Pan

College of Computer Science and Engineering
Shandong University of Science and Technology
Qingdao 266590, China
School of Computer Science and Mathematics
Fujian University of Technology
Fuzhou, 350118, China
Department of Information Management
Chaoyang University of Technology
Taichung, 413310, Taiwan
jengshyangpan@gmail.com

Jiawen Zhuang

School of Computer Science and Mathematics
Fujian University of Technology
Fuzhou, 350118, China
1085231304@qq.com

Lyuchao Liao

School of Computer Science and Mathematics
Fujian University of Technology
Fuzhou, 350118, China
fjachao@gmail.com

Shu-Chuan Chu*

College of Computer Science and Engineering
Shandong University of Science and Technology
Qingdao 266590, China
scchu0803@gmail.com
(Corresponding author: Shu-Chuan Chu)

Received January 2021; revised March 2021

Abstract. *Electric Vehicle Routing Problem with Time Windows (EVRPTW) is often used in the transportation area. EVRPTW is difficult to solve by the traditional precise method. The meta-heuristic algorithm is often used to solve EVRPTW and can obtain approximate optimal solutions. Equilibrium optimizer (EO), as a meta-heuristic algorithm, is simple to implement by software and hardware. Given the shortcomings of EO, such as low convergence precision and fall into local optima easily, we propose an advanced equilibrium optimizer (AEO). In AEO, we improved EO with multi-population method, novel quantum operator, and FPA-inspired pollination operator. Multi-population method constitutes the algorithm structure of AEO. The novel quantum operator and FPA-inspired pollination operator effectively enhance EO's global exploration capabilities, improving the convergence accuracy and stability of EO. Then we test the AEO by CEC2013. Experiment results and Friedman's mean rank show that AEO has better performance in convergence than differential evolution (DE), flower pollination algorithm (FPA), grey wolf optimizer (GWO), particle swarm optimization (PSO), and EO. Finally, AEO also is applied to solve EVRPTW. From the test results of the instances, AEO is more suitable to solve the EVRPTW than some comparison algorithms.*
Keywords: meta-heuristic algorithm, equilibrium optimizer, CEC2013, electric vehicle routing problem with time windows.

---

1. **Introduction.** The meta-heuristic algorithm is widely used in transportation problems. One of the meta-heuristic algorithms is population-based algorithm [1, 2]. Differential Evolution (DE) [3, 4], Cuckoo Search (CS) [5, 6], PSO [7, 8], artificial bee colony (ABC) algorithm [9, 10], FPA [11, 12], Grey Wolf Optimization (GWO) [13, 14] and Quasi-Affine Transformation Evolutionary Algorithm (QUATRE) [15, 16] were some popular population-based algorithms. EO, as a novel swarm intelligence algorithm, is proposed by Faramarzi [17]. The idea of ABC comes from the gather honey behavior. Fish Migration Optimization (FMO) [18] is implemented based on the migration characteristics of the fish. FMO has been applied to several applications [19, 20]. The enlightenment of EO comes from the control volume mass balance model.

EO was introduced in 2020, some researchers have studied it. Wunnava proposed adaptive equilibrium optimizer for multilevel thresholding in computer vision [21]. Gao et al. presented two binary equilibrium optimizer algorithm and for selecting the optimal feature subset [22]. Abdul-hamied et al. uses EO to solve the optimal power flow problem [23]. Rabehi uses EO to optimal estimation of schottky diode parameters [24]. Nusair uses EO to optimal power flow problem with high penetration of renewable energy, and it provides the lower optimization value in term of electric power generation, real power loss, emission index and voltage deviation [25].

The performance of many original swarm intelligence algorithms is not so satisfactory in many applications, result in researchers have adopted many strategies for improvement.

Multi-population method is a simple way to make algorithm to avoid the local optima. There are many researchers pay close attention to this method. PSO is improves with multi-population with three communication strategies, has been tested by some optimization function with prospective results [26]. The ant colony optimization also use multi-population and is improved by using some communication methods and is applied for solving the TSP with promising solutions [27]. The cat swarm optimization (CSO), proposed by Chu et. al., has been widely used in numerous fields, and the multi-population method is also used in CSO, which gets better results and is applied to several problems [28, 29, 30, 31, 32].

Many scholars have also studied the combination of intelligent computing and quantum computing. Sun established a novel PSO with quantum behavior by quantum Delta potential well model [33, 34]. Similarly, Lu introduced the quantum operator into the global pollination of FPA and named it as QFPA [35].

Although VRP was proposed by Dantzig in 1959, it has been many years [36], as a branch of VRP, the related research of electric vehicle routing problem (EVRP) started relatively late [37].

EVRP is an important issue. On the one hand, it's a practical problem, which uses electric vehicles (EVs), EVs are powered by electricity and have the advantages of zero greenhouse gas emission, low noise and strong maintainability. There are logistics services and transportation companies that intend to use EVs for cargo distribution tasks [38]. On the other hand, EVRP is also a challenging scientific issue. Because the vehicle adopts EV, it is more complex than traditional VRP that is NP-hard problem [39], and more factors need to be considered, which makes it more difficult to solve.

Many scholars have studied EVRP. It can be regarded as a variant of GVRP [40], and it mainly involves alternative fuel vehicles, which have a limited range and should be recharged on the way [41]. Schneider introduce the EVRP with time windows and recharging stations, its charging can be done at any available charging station using an appropriate charging scheme [42]. Kancharla introduces a three-index formulation for EVRP with non-linear charging and load-dependent discharging [43]. Gao establishes a EVRPTW model with penalty function, uses genetic algorithm to solve, and genetic operators are designed [44].

In this paper, AEO is proposed, that improved with multi-population method, novel quantum operator, and FPA-inspired pollination operator. Then the algorithms are tested by CEC2013 functions [45], and AEO has better convergence precision and stability than DE, FPA, GWO, PSO, and EO. We also applied these algorithms to solve EVRPTW. From the experimentation of the EVRPTW, AEO achieved good results.

The following is the remaining of this paper. In section 2, the EVRPTW model is described. Section 3 introduces EO. AEO is described in section 4. In section 5, the experiments of CEC2013 functions and EVRPTW are described. In section 6, a conclusion is given.

2. EVRPTW.

EVRPTW is that using EVs to deliver goods from the depot to customer with time windows, in which EVs need to be charged, and its goal is to minimize the total distance in this research.

There are some assumptions in EVRPTW. Each EV can enter each charging station at most once. Each EV is fully charged through the charging station. All EVs have the same and constant speed. The flow of goods is one-way, pure delivery without collection. The departure time of the EV from the depot is 0 o'clock. All EVs start and end with the depot. The energy of EVs is restricted by the driving range, and the driving range of EVs is known. The EVRPTW model is as follows [44].

$$\min f = Cost_{\text{distance}} \sum_{k\in K_{Veh}} \sum_{i\in V_{point}} \sum_{j\in V_{point}} x_{ij}^{k} d_{ij} + \sum_{k\in K_{Veh}} \sum_{i\in N_{cus}} penaltyFunc_i(t_i) \tag{1}$$

s. t.

$$\sum_{k\in K_{vel}} \sum_{i\in N_{cus}} x_{oi}^{k} = \sum_{k\in K_{vel}} \sum_{j\in N_{cus}} x_{jo}^{k} \tag{2}$$

$$\sum_{i\in N_{cus}, i\neq j} x_{ij}^{k} = \sum_{j\in N_{cus}, i\neq j} x_{ij}^{k} = y_i^k, \forall k \in K_{vel} \tag{3}$$

$$\sum_{i \in N_{cus}} y_i^k q_i \leq Q, \forall k \in K_{vel} \tag{4}$$

$$\sum_{k \in K_{vel}} \sum_{i \in V_{point}, i \neq j} x_{oi}^k \leq |K_{vel}| \tag{5}$$

$$\sum_{i \in N_{cus}} \sum_{j \in V_{point}, j \neq i} x_{ij}^k \leq |N_{cus}|, \forall k \in K_{vel} \tag{6}$$

$$t_o^2 = 0 \tag{7}$$

$$t_{ij} = \frac{d_{ij}}{speed}, \forall i, j \in V_{point} \tag{8}$$

$$t_i^2 = t_i^1 + tf_i + tw_i, i \in N_{cus} \cup M_{station} \tag{9}$$

$$t_j^1 = \sum_{i \in V_{point}} \sum_{j \in V_{point}, i \neq j} x_{ij}^k (t_i^2 + t_{ij}), \forall k \in K_{vel} \tag{10}$$

$$tw_i = \max[0, (e_i - t_i^1)], \forall i \in N_{cus} \tag{11}$$

$$p_{ik}^1 = p_{ik}^2, \forall i \in N_{cus}, \forall k \in K_{vel} \tag{12}$$

$$p_{ik}^2 = P, \forall i \in O \cup M_{station}, \forall k \in K_{vel} \tag{13}$$

$$p_{vk}^1 \geq 0, \forall v \in V_{point}, \forall k \in K_{vel} \tag{14}$$

$$x_{ij}^k, y_i^k \in \{0, 1\}, \forall i, j \in V_{point}, \forall k \in K_{vel} \tag{15}$$

$$penaltyFunc_i(t_i^1) = EarlyP \times \max(e_i - t_i^1, 0) + LaterP \times \max(t_i^1 - l_i, 0) \tag{16}$$

where N_{cus} is the of customers number, and K_{vel} is the number of vehicle, $M_{station}$ is the number of charging station, O is the depot, V_{point} is the point set that consist of customers, stations and depot, $Cost_{\text{distance}}$ is the transportation cost of per unit distance of EV. d_{ij} is the distance of traveling from the ith customer to jth customer, q_i is the demand of the ith customer. The capacity of the EV is Q. p_{ik}^1 is remaining power when the kth EV reaches the ith customer. p_{ik}^2 is remaining power when the kth EV leaves the ith customer. P is the energy of power of EV. e_i is ith customer's early arriving time windows. l_i is ith customer's later arriving time windows. $EarlyP$ is the penalty of early arriving. $LaterP$ is the penalty of later arriving. t_i^1 is time for EV to reach ith customer. t_i^2 is time for EV to leave ith customer. tw_i is waiting time for EV at ith customer. tf_i is service time of EV at ith customer or charging time at ith charging station. t_{ij} is time required for an EV to go from ith point to jth point. $speed$ is speed of EV. If the kth EV goes from ith point to jth point, x_{ij}^k equal to 1, otherwise, x_{ij}^k equal to 0. If ith customer is serviced by the kth EV, y_i^k equal to 1, otherwise, y_i^k equal to 0 [44].

Equation (1) is the fitness function that is to minimize the total distance and total time penalty function. Equation (2) ensure that the starting and ending points are depot. Equation (3) guarantees that each customer can be served only once. Equation (4) is capacity limitation. Equation (5) indicates that the EV used in the service does not exceed the maximum number. Equation (6) indicates that the number of customers served by each EV should not exceed the number of customers. Equation (7) shows the departure time of the EV. Equation (8) is the calculation of travelling time. Equation (9) indicates that the time for the EV to leave ith customer is equal to the sum of the time for the EV to arrive at ith customer, the service time at ith customer, and the waiting time at ith customer. Equation (10) is the calculation of jth customer's arriving time that is the sum of ith customer's leaving time and travelling time from ith customer to jth customer. Equation (11) is the calculation of waiting time. Equation (12) ensures that EVs do not consume energy during their stay at customer. Equation (13) indicates that the EV is fully charged when it leaves the depot or changing station. Equation (14) ensures that EVs have enough energy to leave. In (15), x_{ij}^k is decision variable. Equation (16) is time penalty function [44].

In this paper, demand constraint and energy constraint are regarded as penalty function and put them into fitness function to facilitate program design. The following is the final fitness function [44].

$$\begin{aligned} \min f = Cost_{\text{distance}} & \sum_{k\in K_{vel}} \sum_{i\in V_{point}} \sum_{j\in V_{point}} x_{ij}^k d_{ij} \\ & + \sum_{k\in K_{vel}} \sum_{i\in N_{cus}} penaltyFunc_i(t_i) \\ & + \sum_{k\in K_{vel}} M_L \max(\sum_{i\in N_{cus}} y_i^k q_i - Q, 0) \\ & + \sum_{k\in K_{vel}} M_L \max(\sum_i \sum_j x_{ij}^k d_{ij} - DisLimit, 0) \end{aligned} \tag{17}$$

where M_L is large number for penalty, and $DisLimit$ is the driving range limitation, representing the energy constraint.

3. **EO.** EO, as a novel population-based and physical-based algorithm, is inspired by the control volume mass balance model. EO initialize concentrations like population-based algorithms, the following is EO's initialization.

$$C = C_{\min} + rand \times (C_{\max} - C_{\min}) \tag{18}$$

where $rand$ is a random vector, and obeys uniform distribution from 0 to 1. A mass balance equation, is the core updating equation of EO. The following is balance equation.

$$V\frac{dC}{dt} = QC_{eq} - QC + G \tag{19}$$

where C is the concentration inside the control volume V, and V is considered as a unit. $V\frac{dC}{dt}$ is the rate of change of mass. Q is the volumetric flow rate. C_{eq} is the equilibrium state concentration. G is mass generation rate. Using turnover rate λ to rearrange mass equation. λ is calculated as follows.

$$\lambda = Q/V \tag{20}$$

To integrate the mass equation, from C_0 to C and from t_0 to t. C_0 and t_0 are the start time and initial concentration. The following is the result equation, or the updating equation.

$$C = C_{eq} + F(C_0 - C_{eq}) + (1 - F)G/(\lambda V) \tag{21}$$

In equation (21), there are three parts. The first is equilibrium state concentration that is selected from equilibrium pool. The second is about global searching, with respect to the difference between a solution and the equilibrium concentration. The third is associated with generation rate, is often act as exploiter, and sometimes is act as explorer.

The equilibrium pool includes five solution, which are first four better solution and a mean solution of mentioned four solution. The following is equilibrium pool.

$$C_{eq,pool} = \{C_{eq1}, C_{eq2}, C_{eq3}, C_{eq4}, C_{eq,ave}\} \tag{22}$$

Exponential term F balances between exploration and exploitation. The following is the calculation of F.

$$F = e^{-\lambda(t-t_0)} \tag{23}$$

where λ is random vector, obeys uniform distribution from 0 to 1. t is the function of generation. t and t_0 are calculated as follows.

$$t = (1 - \frac{gen}{Maxgen})^{(a_2 \frac{gen}{Maxgen})} \tag{24}$$

$$t_0 = t + (\ln(-a_1 \times sign(r - 0.5) \times (1 - e^{-\lambda t})))/\lambda \tag{25}$$

where gen and $Maxgen$ are the current and maximum generation, $sign(r - 0.5)$ controls the direction of exploration and exploitation. r is a random vector similar to λ. It is suggested that $a_1 = 2$ and $a_2 = 1$ in [17]. Then the revised formula about F is as follows.

$$F = a_1 \times sign(r - 0.5) \times (e^{-\lambda t} - 1) \tag{26}$$

In third term, G is calculated by the following equation.

$$G = G_0 e^{-k(t-t_0)} \tag{27}$$

$$G_0 = GCP(C_{eq} - \lambda C) \tag{28}$$

$$GCP = \begin{cases} 0.5r_1 & r_2 \geq GP \\ 0 & r_2 \prec GP \end{cases} \tag{29}$$

where G_0 is the initial value in equation (27), and k is decay constant, $k = \lambda$. r_1 and r_2 are random number and are similar to λ. GCP is generation rate control parameter. $GP = 0.5$ that is suggested by Faramarzi [17]. EO also has the particle's memory saving mechanism. The steps of EO is as follows.

Step 1. Initialize population number (ps), maximum generation ($Maxgen$), dimension of solution (d), $a_1 = 2$, $a_2 = 1$, $GP = 0.5$, $V = 1$, population (C).

Step 2. Calculate fitness of current generation particles. Update C_{eq1}, C_{eq2}, C_{eq3}, C_{eq4} and their fitness.

Step 3. Particles' memory saving.

Step 4. Calculate $C_{eq,ave}$ by $C_{eq,ave} = (C_{eq1}+C_{eq2}+C_{eq3}+C_{eq4})/4$. Establish equilibrium pool. Calculate t using (24).

Step 5. Each particle choose C_{eq} randomly in $C_{eq,pool}$, produce λ and r randomly, calculate exponential term F by (26), calculate GCP, G_0, G by (27)-(29), update by (21).

Step 6. Step 2 to 5 are repeated until reaching the given threshold fitness, or maximum generation. Finally, record global best concentration C_{eq1} and fitness of it.

4. **Proposed Advanced EO.** EO has strong practicability in software and hardware for its simple mass balance equation that is similar to FPA. It also has shortcomings such as low convergence precision and fall into local optimum easily. Given the shortcomings of EO, we use novel multi-population method, novel quantum operator and FPA-inspired pollination operator to promote EO's exploration ability and convergence precision and stability.

4.1. **Multi-population method.** Multi-population method is a simple way to make algorithm to avoid the local optima. In this paper, EO with multi-population method, that is MEO. The population was divided into *groupNum* groups, each group runs EO. Because communication destroys the parallelism of multiple groups and affects the global search capability, in order to avoid this situation and from the point of view of simplicity and practicality, we do not use communication. This constitutes the algorithm framework of AEO.

4.2. **Novel quantum operator.** In [33], Sun et al. proposes PSO with quantum behavior, which uses wave function in quantum mechanism. The wave function is calculated by following formula.

$$i\hbar\frac{\partial\psi(X,t)}{\partial t} = \widehat{H}\psi(X,t) \tag{30}$$

where ψ is wave function, i is imaginary unit, $\hbar$ is the Planck constant, the position vector of the particle is X, and $\widehat{H}$ is a Hamiltonian operator, calculated by the following formula [33].

$$\widehat{H} = -\frac{\hbar\nabla^2}{2m} + V(X) \tag{31}$$

where m is the mass of the particle, $V(X)$ is the potential field, ∇^2 is the Laplace operator. A δ potential well is established with p as the center of the attraction potential and its potential function is $V(X)$. The following is the calculation of $V(X)$ [33].

$$V(X) = -\gamma\delta(X-p) = -\gamma\delta(y) \tag{32}$$

Through above equations, the wave function is obtained as follows [33].

$$\psi(y) = \frac{e^{-|y|/L}}{\sqrt{L}} L = \hbar^2/(m\gamma) \tag{33}$$

The probability density function is calculated as follows [33].

$$PDF(y) = |\psi(y)|^2 = (e^{-2|y|/L})/L \tag{34}$$

The above formula is the formula of quantum space, and we need to convert to the classical space, Monte Carlo method is a way to achieve [33]. Let $u/L{=}PDF(y)$, u is random vector, obeys uniform distribution from 0 to 1, then the value of y is as follows.

$$y = \pm\frac{L\ln(1/u)}{2} \tag{35}$$

where L is evaluated by the following formula [34].

$$L = 2a\,|p-X| \tag{36}$$

where a is a decreasing number [34]. Then the quantum operator is as follows.

$$y = \pm a\,|X-p|\ln(1/u) \tag{37}$$

Through some experimental analysis of F and using MATLAB to draw the trend of F in iteration, it can be concluded that F attenuates from the approximate $[-a_1, a_1]$ range of oscillation to near 0, and its maximum amplitude is less than a_1 in EO. We propose a novel quantum operator, where a is replaced with EO's exponential term (F) and attraction center (p) is as follows.

$$\begin{gathered} p = \zeta_1 Pbest + \zeta_2 C_{eq1} + \zeta_3 C_{eq} + \zeta_4 C_{eq3} + \zeta_5 C_{eq4} \\ \zeta_1+\zeta_2+\zeta_3+\zeta_4+\zeta_5 = 1 \end{gathered} \tag{38}$$

where $Pbest$ is the mean vector of total particle's history best solution, ζ_1 to ζ_5 is random number, obeys uniform distribution from 0 to 1, and the sum of them equals to 1. EO with novel quantum operator is QEO. The revised updating equation in QEO is as follows.

$$\begin{gathered} C = C_{eq} + F(C_0 - C_{eq}) + (1-F)G/(\lambda V) \\ \pm F\,|\mathrm{C} - p|\ln(1/u) \end{gathered} \tag{39}$$

4.3. **FPA-inspired pollination operator.** The FPA has two processes: global pollination and local pollination. It uses switching probability to switch pollination methods. Global pollination uses Levy flight to explore, and local pollination uses particle difference to exploit. We design an FPA-inspired pollination operator for EO updating equation, that effectively utilize the advantages of FPA. Based on the FPA-inspired pollination operator, the novel EO is FEO. The following is the updating equation with FPA-inspired pollination operator [11, 12].

$$\begin{gathered} C = C_{eq} + F(C_0 - C_{eq}) + (1-F)G/(\lambda V) \\ +t * \gamma * Levy(Dim_{current}) * \mu_{\mathrm{rand}} * (C_m - C_n) \end{gathered} \tag{40}$$

where the last part in (40) is FPA-inspired pollination operator. γ is the scaling factor. $Levy(Dim_{current})$ is random number based on Levy flight, which reflects the global pollination capacity of the FPA. μ_{rand} is a random vector, obeys uniform distribution from 0 to 1. $C_m - C_n$ is the particle difference of two different randomly selected particle. $\mu_{\mathrm{rand}} * (C_m - C_n)$ reflects the local pollination capacity of the FPA. t is calculated by (24), and it is the oscillation decays to 0, we add it to this operator to improve the convergence ability of it.

4.4. **AEO.** In this paper, we have designed a AEO that combines the multi-population method, novel quantum operator and FPA-inspired pollination operator. Not only the global search capability of the multi-population method and novel quantum operator is effectively utilized, but also the pollination optimization capability of FPA is integrated, which further improves the convergence performance of EO. The following is the updating equation of AEO.

$$\begin{gathered} C = C_{eq} + F(C_0 - C_{eq}) + (1-F)G/(\lambda V) \\ \pm F\,|\mathrm{C} - p|\ln(1/u) \\ +t * \gamma * Levy(Dim_{current}) * \mu_{\mathrm{rand}} * (C_m - C_n) \end{gathered} \tag{41}$$

For AEO, we set the total population is $groupNum \times ps$. Firstly, concentrations are divided into $groupNum$ groups. Each group updating EO with (41). While meeting the max number of iteration or calculate fitness is just less than the threshold fitness, terminate the AEO. The following are the detailed steps.

Step 1. Initialization: The population is divided into $groupNum$ groups, each group runs EO independently. Produce the mth group's ps concentrations C_m^{gen} with $Dim_{current}$ dimensions, where ps is the size of population, gen is the present generation and set $gen = 1$. To initialize maximum generation $Maxgen$, a_1, a_2, GP, V, γ, total best solution $gbest$ and total best value $gbestval$. Assign each group's equilibrium candidates' fitness and total best value $gbestval$ a large number. Each group's C_{eq1}, C_{eq2}, C_{eq3}, C_{eq4} and $gbest$ are set to zero vector.

Step 2. Evaluation: Calculate each group's current concentration fitness $f(C_m^{gen})$. Update the mth group's $C_{m,ceq1}^{gen}$, $C_{m,ceq2}^{gen}$, $C_{m,ceq3}^{gen}$ $C_{m,ceq4}^{gen}$ and corresponding fitness. Update total best solution $gbest$ and total best value $gbestval$.

Step 3. Particle's history best update: If $gen > 1$, update each group's particle history best solution $Pbest$ and fitness $Pbestfit$. According to (38) to calculate attraction center p.

Step 4. Memory saving: If $gen > 1$, each group's particle memory saving mechanism is used.

Step 5. Each group's EO updating: Calculate each group's average concentration $C^{gen}_{m,eq,ave}$. Establish each group's equilibrium pool. Calculate each group's t by (24). Updating: Select $C^{gen}_{m,eq}$ from each group's equilibrium pool, generate λ and r randomly, calculate each group's exponential term F, GCP, G_0, G. Update each group's concentrations by (41).

Step 6. Termination: Step 2 to 5 are repeated until reaching the given threshold fitness, or maximum generation. Finally, record the total best fitness *gbestval* and total best solution *gbest* among the concentrations.

4.5. **Setting for EVRPTW.** Both the EO and the AEO are continuous optimization algorithms, which are often used in continuous optimization problems.

In this paper, for EVRPTW, we adopt n + m method [46] for solution representation, means route is n + m dimensions, that is m vehicles and n customers, and the path is the sequence of numbers between the two zeros in the solution, as is shown in figure 1, there are three ways, 0-4-2-3-0, 0-1-6-0, 0-5-0. 0 is the depot point, is the start and end of a path.

0	4	2	3	0	1	6	0	5	0

FIGURE 1. n + m method.

For continuous solution to discrete route solution, we design a novel encoding way for EVRPTW. Suppose there are m vehicles, n customers and c charging stations. The solution of the continuous algorithm is set as n+c dimension, and the value of each dimension is range from 1 to m + 1. After rounding each dimension, the result is the path to which the customer or charging station is assigned. In the same path, sort from small to large according to the corresponding continuous value of each dimension, and the result is the final path. As is shown in figure 2. There are 3 vehicles, 4 customers and 2 charging stations that are Point 2 and 6. According to encoding, the final paths are 0-4-2-3-0, 0-1-6-0, 0-5-0.

Similarly, we designed the corresponding decoding method. Suppose we have the solution shown in figure 1. First, the solution is allocated by vehicle, from front to back, the first path, the second path, and so on. Then, randomly generate random numbers that follow a uniform distribution from 0 to 1, and sort them from small to large. Finally, the number obtained by adding the random number to the vehicle serial number is the value corresponding to the point value. The decoding steps are shown in figure 3.

To further improve the quality of the solution, we optimize the worse solution that exceeds the energy constraint, that is local searching. Traverse the population, randomly pick two solutions one after another without repeating. If there is a worse solution that exceeds the power constraint, local searching method will be performed, which is inspired from [44]. But we only use crossover operator [44] to produce new solutions within worse solution, and select the better solution in the end. With the iteration, the number of worse solutions decreases and the number of runs of local searching decreases, which is different from the operation mechanism of crossover operator in [44]. We also optimize the solution that not exceeds the energy constraint, with certain probabilities. To switch two points in a path of a solution in turn, if the new one is better, than replace it.

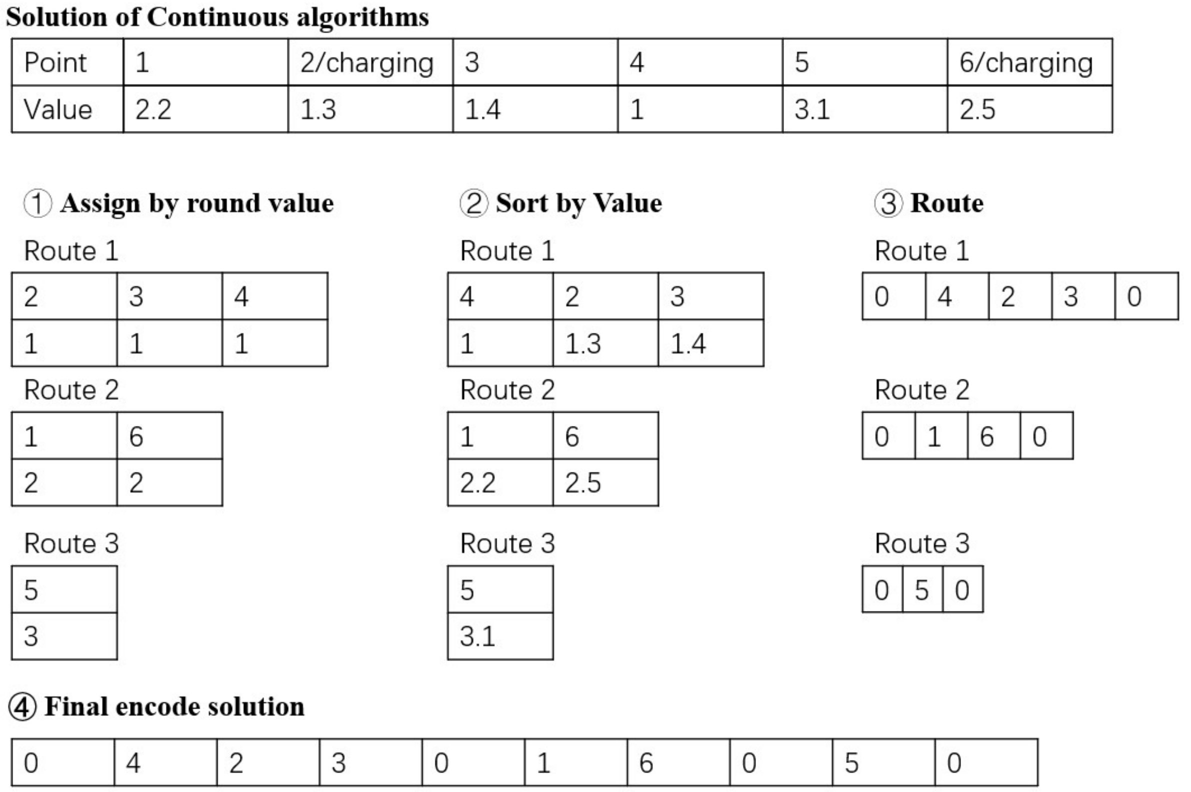

Figure 2. Encoding way for EVRPTW.

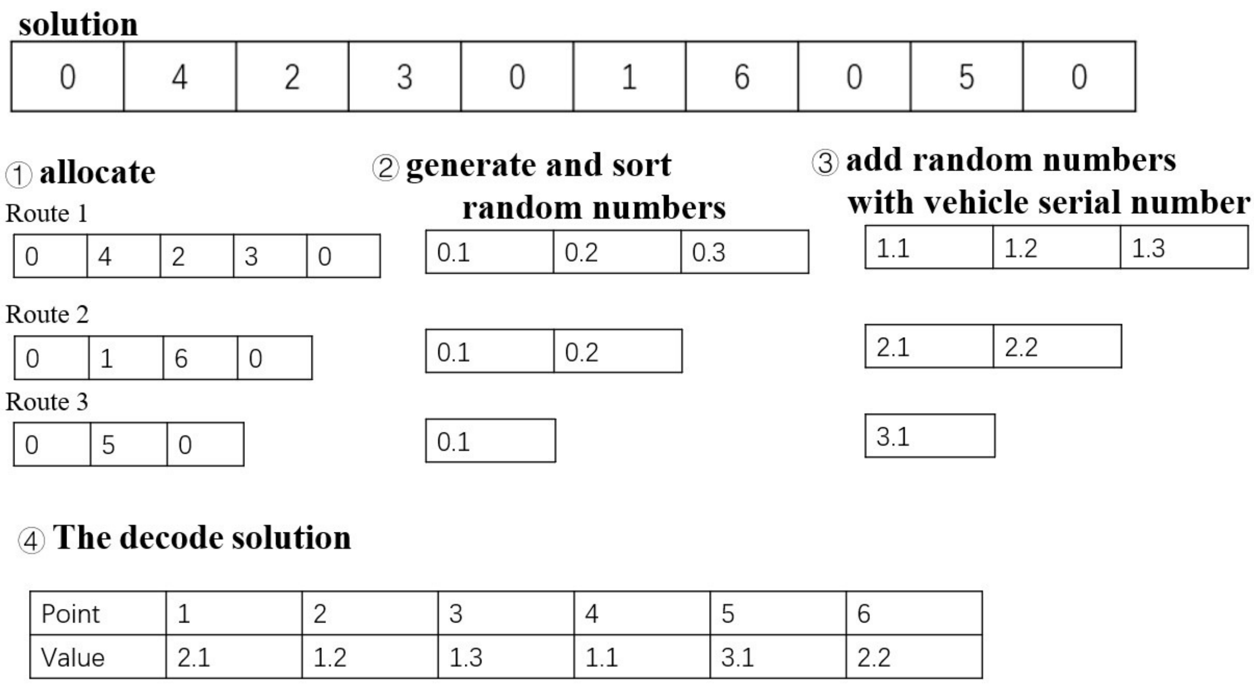

Figure 3. Decoding way for EVRPTW.

5. **Experiment and Application.** In this section, we utilized CEC2013 to test our proposed algorithm, detail function is shown in table 1. First, AEO results show in table 3 to 5. The convergence curves of AEO show in figure 4 to 6, which compare with DE, FPA, GWO, PSO, and EO. Then, we fixed the number of iterations to compare the performance of two AEO operators and multi-population method in different dimensions, detail results are shown in table 6 to 8. Finally, we apply AEO to solve the EVRPTW, the result is as shown in table 9 and figure 7.

TABLE 1. Function of CEC2013

No.	Type	Optimum	No.	Type	Optimum
F1	Unimodal	-1400	F15	Multimodal	100
F2	Unimodal	-1300	F16	Multimodal	200
F3	Unimodal	-1200	F17	Multimodal	300
F4	Unimodal	-1100	F18	Multimodal	400
F5	Unimodal	-1000	F19	Multimodal	500
F6	Multimodal	-900	F20	Multimodal	600
F7	Multimodal	-800	F21	Composition	700
F8	Multimodal	-700	F22	Composition	800
F9	Multimodal	-600	F23	Composition	900
F10	Multimodal	-500	F24	Composition	1000
F11	Multimodal	-400	F25	Composition	1100
F12	Multimodal	-300	F26	Composition	1200
F13	Multimodal	-200	F27	Composition	1300
F14	Multimodal	-100	F28	Composition	1400

5.1. **Experiment setting.** Three types of functions are included in CEC2013, as shown in table 1. The first type is unimodal function, which test the exploitation ability. The second is the basic multimodal function, which test the exploration ability. The third type is composition function, representing challenging problems. The search range is [-100,100].

Table 2 shows the setting of algorithms. For example, as for AEO, its population is divided into $groupNum$=4 groups, each group update EO with (41), $a_1 = 2$, $a_2 = 1$, $GP = 0.5$, $V = 1$, γ=0.1. In this paper, each algorithm has 100 particles that is ps=100, with $Dim_{current}$. Each algorithm has 31 independent runs in each benchmark, and $Maxgen$ equal to $10000 \times Dim_{current}/ps$. The dimension in this section are 10, 30, 50.

Qualitative metric use convergence curve, the quantitative measure comprises of the mean and standard deviation values of the specific benchmark functions. We also use Friedman's mean rank in analysis.

TABLE 2. Setting of each algorithm

Algorithm	Parameter
DE	$F = 2$, $CR = 0.9$, use DE/rand/1/bin
FPA	$p = 0.8$
GWO	α decreases linearly from 2 to 0
PSO	$c_1 = c_2 = 2$,$\omega = 0.8$
EO	$a_1 = 2$, $a_2 = 1$, $GP = 0.5$, $V = 1$
AEO	$groupNum$=4, $a_1 = 2$, $a_2 = 1$, $GP = 0.5$, $V = 1$
	γ=0.1, with novel quantum operator

5.2. **AEO experiment.** Table 3 to 5 are experiment result of AEO with DE, FPA, GWO, PSO, and EO. We use fitness error $F - F_{optimum}$ for simplicity. The mean item is the mean value of 31 runs and the std item is stand deviation. The UniNum item records the best number of each algorithm in unimodal functions, including mean and std item. The MulNum item is about multimodal functions, the ComNum item is about composed functions, the Total item is about total functions. In each table, if an algorithm achieves the best of the four algorithms in a function test, the number of records is increased by one. Note that two algorithms get the same best results in the same function, we record both algorithms getting the best results. The final line is the Friedman's mean rank. figure 4 to 6 are the convergence curves of AEO that compare to DE, FPA, GWO, PSO, and EO.

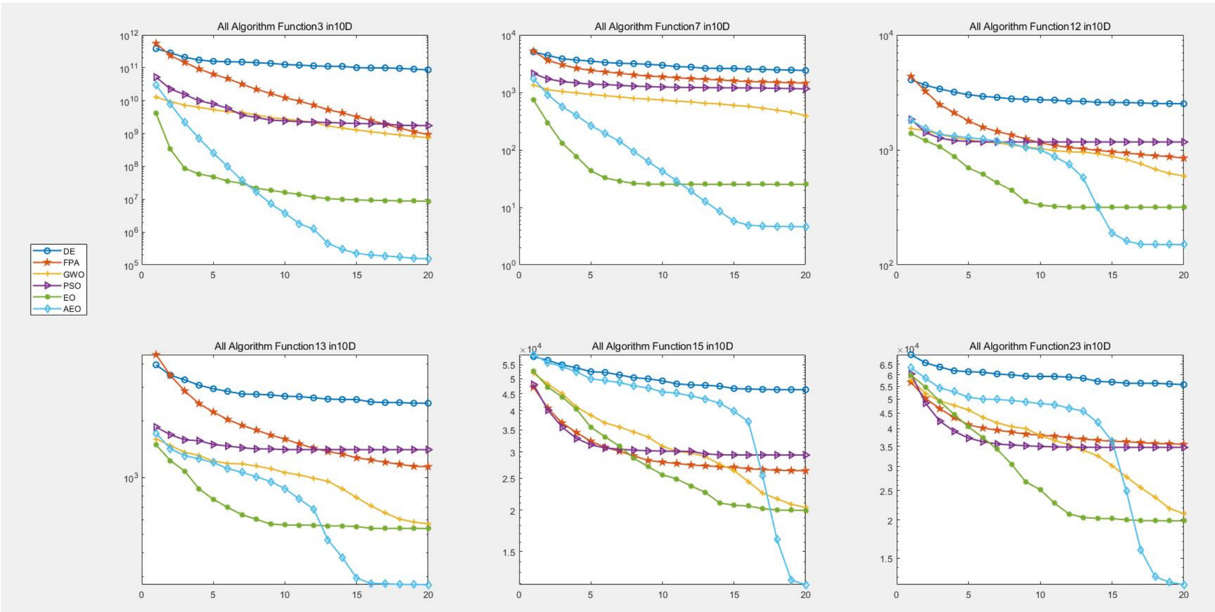

FIGURE 4. The convergence curve of AEO with the dimension of 10.

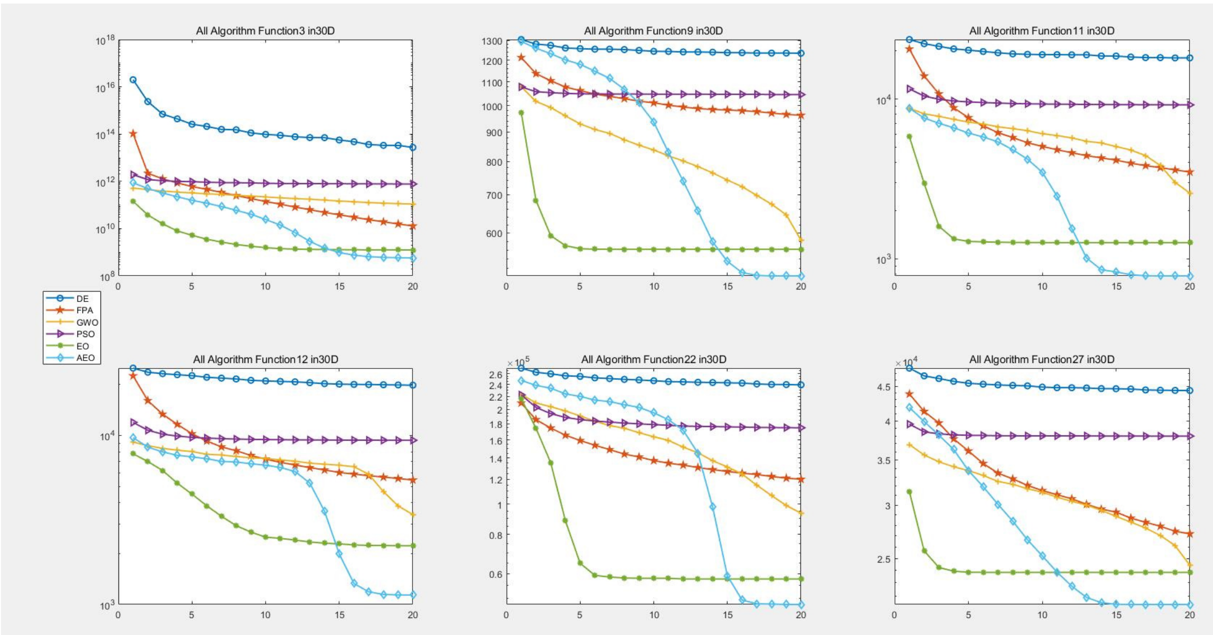

FIGURE 5. The convergence curve of AEO with the dimension of 30.

As shown in table 3 to 5, AEO has a better mean result, mainly multimodal functions and composition functions, proving that AEO's global searching ability, or exploration ability, is better than DE, FPA, GWO, PSO and EO. AEO is also more applicable to

Table 3. Experiment result of AEO with the dimension of 10

Fun	Item	DE	FPA	GWO	PSO	EO	AEO
F1	mean	1374.7048	0.053809	4.4232	9.5959e-11	4.4008e-14	2.2004e-13
	std	313.238	0.01836	24.6015	1.9884e-10	9.1315e-14	1.0958e-13
F2	mean	10544369.19	591.8552	1124842.676	125.3745	96091.217	255847.5315
	std	3892127.833	250.8412	1055318.053	261.3769	80849.9335	177156.0021
F3	mean	2796114774	30073804.36	24320736.08	55948289.19	279693.1813	4947.9666
	std	1002927427	13448862.81	41947509.96	123494921.8	660114.9891	8765.0567
F4	mean	13873.873	25.7877	7108.4316	0.583	524.6526	2791.1638
	std	4288.9387	9.1141	3922.2683	2.4465	579.6013	1444.908
F5	mean	79.8038	0.2651	27.4086	0.0020148	8.0681e-14	3.2738e-10
	std	15.4533	0.055204	40.7751	0.0041672	6.6895e-14	3.6243e-10
F6	mean	107.2844	0.31426	11.1103	1.1373	7.9435	5.4304
	std	23.1788	0.27676	3.0177	1.6778	3.8794	4.503
F7	mean	77.551	46.6776	12.7381	37.3117	0.81536	0.14931
	std	12.2059	9.3871	9.9732	26.4922	1.0688	0.11302
F8	mean	20.3495	20.3569	20.3554	20.3757	20.3646	20.352
	std	0.060131	0.082015	0.048784	0.093069	0.062883	0.07739
F9	mean	8.8429	5.9584	3.6059	5.803	2.6679	1.7061
	std	0.60593	0.55605	1.0774	1.4028	1.2675	0.81089
F10	mean	181.0785	0.18657	13.4482	1.0407	0.17588	0.093161
	std	47.9272	0.034792	18.1375	0.70518	0.12459	0.044701
F11	mean	69.7613	19.9561	10.7997	30.0092	1.8615	3.2738
	std	8.2238	3.5023	6.2566	14.6972	1.1979	1.1544
F12	mean	81.3502	27.4048	19.1764	37.8404	10.2385	4.8472
	std	10.0944	4.6673	10.7149	18.9328	4.1836	1.4921
F13	mean	79.9942	36.8926	18.3952	45.4029	17.3558	8.7482
	std	10.5017	7.1331	11.4278	12.9821	7.7278	5.4131
F14	mean	1300.2895	527.73	423.7707	975.9461	182.224	93.2127
	std	167.9756	85.551	172.4106	333.3272	113.1252	85.0212
F15	mean	1492.907	849.4663	657.8551	947.6495	644.967	384.6483
	std	160.5207	125.8852	360.2377	258.604	281.6923	155.2052
F16	mean	1.2231	0.90088	1.2154	0.86934	0.60906	0.77288
	std	0.18941	0.18444	0.17676	0.41051	0.1719	0.20621
F17	mean	128.8709	40.9636	24.5781	36.8558	16.8563	14.7475
	std	14.8429	5.5259	5.6901	15.9195	2.6489	1.9235
F18	mean	129.1563	48.4575	35.8459	32.9589	19.2734	16.5906
	std	14.3909	6.0855	6.3393	10.6848	3.4259	2.4012
F19	mean	36.0978	1.4664	1.4689	1.5495	0.80716	0.57174
	std	18.1006	0.30492	0.74407	0.90256	0.15777	0.16184
F20	mean	3.9117	3.4407	2.6349	3.4647	2.237	2.1458
	std	0.14912	0.27995	0.45094	0.4662	0.53279	0.33868
F21	mean	517.8119	121.5315	395.4186	400.1939	400.1939	393.736
	std	22.5764	24.1267	20.2956	1.2357e-10	2.607e-13	35.9559
F22	mean	1621.7517	706.0613	685.0657	1243.5796	250.3869	229.7843
	std	161.3155	101.4305	366.337	323.1657	152.9049	116.1451
F23	mean	1798.7377	1144.5292	676.6309	1118.3144	641.7291	395.8764
	std	185.2186	122.3121	382.6621	393.2997	332.8306	171.0717
F24	mean	221.5523	166.1381	211.0775	216.9152	210.4313	206.845
	std	11.8047	19.4797	5.2553	2.908	5.9342	3.1827
F25	mean	218.9777	201.2101	209.9019	217.7796	206.6059	206.0171
	std	12.8599	23.3701	5.8695	2.8067	5.2027	3.5391
F26	mean	195.5733	138.3252	168.5333	193.1263	186.4499	140.1601
	std	8.8689	11.4681	56.4514	58.8979	50.5581	45.2069
F27	mean	634.8766	413.3155	409.9323	481.6753	420.0838	433.2265
	std	23.7851	4.615	91.7292	32.3206	105.9459	92.1518
F28	mean	695.9679	217.1792	343.8405	531.5128	336.3058	293.5484
	std	119.3343	125.5884	117.6071	250.085	107.5282	35.9211
UniNum	mean	0	0	0	2	2	1
	std	0	1	0	1	2	1
MulNum	mean	1	1	0	0	2	11
	std	1	4	1	0	2	7
ComNum	mean	0	5	1	0	0	2
	std	1	3	0	2	1	1
Total	mean	1	6	1	2	4	14
	std	2	8	1	3	5	9
Friedman's		5.8214	3.25	3.6071	4.1607	2.4107	1.75

TABLE 4. Experiment result of AEO with the dimension of 30

Fun	Item	DE	FPA	GWO	PSO	EO	AEO
F1	mean	32563.9821	0.0094711	804.8836	235.4749	4.6208e-13	1.52e-10
	std	3075.7389	0.0029417	756.5982	184.0929	1.4949e-13	1.016e-10
F2	mean	377065283.9	15875.1214	21339739.97	4772494.948	1915013.893	4925931.254
	std	63979623.53	6361.6219	12676214.09	3738862.559	880287.6005	1921969.759
F3	mean	877720000000	417862667.7	3541730492	24929823028	40304887.13	18655426.73
	std	1085890000000	179854669.6	2288887231	13707652850	47884029.03	17007251.65
F4	mean	75814.3883	442.5998	26455.0132	22970.2798	1572.3373	10910.36
	std	9383.4269	156.2499	7898.0456	7497.6967	1091.0111	3174.3236
F5	mean	4312.8668	0.12523	786.8169	230.4296	5.0242e-13	4.3217e-06
	std	572.3914	0.034537	620.2571	89.5173	1.0491e-13	2.9006e-06
F6	mean	3427.7226	15.6802	117.478	107.2949	27.8708	20.8729
	std	512.0018	4.4741	34.3224	39.3188	23.975	5.456
F7	mean	756.7825	118.5728	50.1119	215.5989	19.0665	12.7963
	std	505.419	18.3299	14.169	158.1802	10.969	4.1112
F8	mean	20.9384	20.9506	20.9442	20.9607	20.9519	20.9367
	std	0.050271	0.04795	0.061815	0.068069	0.049151	0.055683
F9	mean	39.8245	31.0447	18.7969	33.7311	18.1277	16.2923
	std	1.026	1.1927	2.4249	2.6625	4.3308	2.4301
F10	mean	4021.1802	0.10457	228.648	117.5674	0.11511	0.21405
	std	683.0068	0.017269	107.2997	71.2797	0.049862	0.079384
F11	mean	583.1246	112.5291	82.9859	296.3861	40.6586	25.1634
	std	32.8234	14.1916	35.5571	55.4211	12.7632	5.6003
F12	mean	639.9562	175.7456	109.6984	302.0713	71.7865	36.681
	std	32.2845	31.5957	46.2664	71.8579	28.1277	8.3175
F13	mean	615.6464	226.4638	172.1684	388.2378	141.5401	82.1211
	std	49.0802	27.4243	45.8747	67.461	33.6953	16.42
F14	mean	6924.5664	3151.5464	2762.9645	4712.7816	2228.8081	1862.8319
	std	284.1567	169.1811	926.729	902.9294	534.5979	375.171
F15	mean	7427.7413	4479.6192	3668.5356	4785.5117	4064.148	3164.6559
	std	293.9629	305.0847	1494.6789	923.7675	766.6372	412.0936
F16	mean	2.4413	2.2813	2.4945	0.07366	1.296	1.3646
	std	0.31593	0.26035	0.3018	0.41012	0.41378	0.30095
F17	mean	1274.4674	204.7358	154.0033	344.8299	71.6189	63.9002
	std	83.8885	27.8515	43.6883	57.1651	12.4119	6.0117
F18	mean	1249.8919	214.714	240.7173	348.6742	109.9939	87.6836
	std	98.9668	27.9451	29.4319	110.4906	21.5581	11.3972
F19	mean	187421.1053	12.4639	15.5373	167.7376	3.3321	2.7356
	std	88853.9347	1.4248	16.3365	232.8266	0.8596	0.40237
F20	mean	14.9234	13.2158	12.1574	14.7765	10.9656	9.7787
	std	0.10435	0.36037	1.5551	0.6139	0.89456	0.64016
F21	mean	3139.1263	216.632	801.1325	626.1239	315.296	248.3876
	std	171.5653	21.8241	308.3967	145.3682	53.2812	50.7999
F22	mean	7720.302	3867.665	3017.4275	5639.3211	1863.5683	1546.1169
	std	321.425	203.1277	1158.2235	949.5908	467.4893	356.4039
F23	mean	7993.638	5538.9125	4121.3536	5601.7054	4123.6915	3216.2494
	std	333.5268	315.3788	1556.0244	744.1271	659.1643	466.2004
F24	mean	334.9318	291.3332	249.7785	292.8641	247.2216	241.8693
	std	4.7212	3.4408	9.5728	6.171	9.8251	6.4956
F25	mean	353.4453	313.6442	270.7647	292.3021	264.1333	253.2047
	std	5.6637	4.3996	9.8791	6.728	7.8832	5.7517
F26	mean	243.247	200.0389	305.2241	353.849	299.4621	204.4823
	std	9.6978	0.024395	68.1141	76.5148	65.4695	23.7329
F27	mean	1431.4658	877.1195	787.5614	1225.0075	768.772	689.014
	std	27.061	306.1689	82.5417	54.6803	97.3533	49.6116
F28	mean	4540.1013	537.5279	1030.7035	2996.1881	293.5484	300.0005
	std	289.7552	61.41	304.7872	699.8849	35.9211	0.00021478
UniNum	mean	0	2	0	0	2	1
	std	0	2	0	0	2	1
MulNum	mean	0	2	0	1	0	12
	std	3	5	0	0	0	7
ComNum	mean	0	2	0	0	1	5
	std	1	6	0	0	0	1
Total	mean	0	6	0	1	3	18
	std	4	13	0	0	2	9
Friedman's		5.7143	3.1786	3.75	4.6071	2.2143	1.5357

TABLE 5. Experiment result of AEO with the dimension of 50

Fun	Item	DE	FPA	GWO	PSO	EO	AEO
F1	mean	79729.4314	0.0031454	3333.5201	8664.0748	1.1369e-12	3.2538e-08
	std	6734.1758	0.0012883	1621.0693	2781.5674	3.1065e-13	2.8045e-08
F2	mean	1192817228	51695.2402	43373232.79	35572236.07	2108725.008	7760673.143
	std	180658005.5	20791.5399	15593326.31	17624860.58	609549.8694	1669389.066
F3	mean	3564830000000	569520969.8	14124582515	52115673255	263259716	330905771.1
	std	2999440000000	214951449.1	4823025376	18592892981	212200209	173871250.1
F4	mean	134914.2548	2390.0014	42413.8922	54920.001	2918.1414	17492.7879
	std	9830.1222	1255.0085	7367.6774	9917.2715	1194.2922	2439.8523
F5	mean	15609.5466	0.060211	853.7383	1004.7113	1.3386e-12	0.00024817
	std	2122	0.016917	328.2674	331.0594	4.6106e-13	8.9731e-05
F6	mean	8258.9788	43.224	236.7663	516.1744	59.1806	44.8244
	std	712.6391	2.8653	57.8431	145.0846	27.9664	2.6846
F7	mean	945.9873	115.7857	60.3723	265.9943	43.5557	33.2521
	std	509.8096	11.2735	12.3975	293.0127	10.9702	6.0935
F8	mean	21.1289	21.1342	21.1366	21.1667	21.1331	21.1177
	std	0.046011	0.035913	0.029466	0.044112	0.032229	0.03622
F9	mean	72.5489	58.1718	37.7561	66.9179	37.1987	36.5144
	std	1.5792	2.0497	4.4157	3.1959	5.8323	3.4712
F10	mean	10198.6574	0.068209	604.2603	916.645	0.14511	1.5266
	std	1049.437	0.015023	238.5814	245.1201	0.05225	0.25813
F11	mean	1329.9465	215.3029	224.3729	634.0813	115.044	55.9352
	std	91.2043	18.9105	50.1113	93.7348	25.8969	6.9541
F12	mean	1330.8731	412.7545	261.8701	654.663	160.8458	83.5633
	std	63.5163	66.1175	93.5276	91.1372	36.5232	13.0495
F13	mean	1310.1615	513.2646	354.4736	804.0346	317.2996	203.7052
	std	74.723	64.0055	71.0953	91.4618	55.3551	28.8016
F14	mean	13335.6669	6626.1335	5197.3439	9806.4756	4285.1139	2972.515
	std	269.1646	479.8601	710.4315	847.5408	928.7505	530.6831
F15	mean	14293.2624	9345.9239	7613.7799	10089.66	8201.2492	6453.5818
	std	309.9198	613.1464	2857.3241	1097.1414	1320.1759	838.7575
F16	mean	3.3469	3.212	3.445	0	1.811	1.6822
	std	0.24458	0.28395	0.25472	0	0.49323	0.33206
F17	mean	3042.7459	379.3733	323.7474	869.9457	172.4191	129.2341
	std	123.9245	45.6791	68.4213	111.0418	25.5675	13.5553
F18	mean	3015.7312	440.7257	515.5844	841.8673	238.6768	183.635
	std	142.866	50.043	34.4932	111.9293	44.9085	17.5186
F19	mean	1375860.703	28.7993	354.7359	2574.7959	7.4643	5.9017
	std	406389.5449	4.7488	444.6428	2032.329	2.4352	0.9328
F20	mean	24.8776	23.3029	20.5598	24.4263	19.9474	19.1664
	std	0.10544	0.60384	1.024	0.6829	0.96862	0.65382
F21	mean	7283.9379	223.8297	2140.5178	2664.6482	937.8353	640.358
	std	196.9782	113.7101	619.774	393.822	138.9779	313.0621
F22	mean	14760.385	8363.9356	6510.3866	11755.8907	4486.755	3232.9546
	std	317.7507	529.0527	1051.9067	1191.5432	951.5458	491.395
F23	mean	15409.0552	11080.6606	7524.3972	11679.9972	8616.2493	6523.1508
	std	350.6766	703.1053	1576.0872	1159.7351	1219.9567	760.2409
F24	mean	464.7022	372.0814	299.3641	375.5314	297.4069	286.6768
	std	10.5763	5.2302	12.6835	7.6889	11.2327	7.6262
F25	mean	501.7526	416.3819	339.7569	375.2746	325.13	306.6817
	std	7.0569	7.8648	11.4522	7.9766	14.6881	8.415
F26	mean	377.803	200.1497	392.1362	454.7906	382.5149	357.4958
	std	23.2689	0.12048	12.8295	67.6374	39.5379	70.3449
F27	mean	2454.857	1751.1033	1275.3574	2046.2024	1291.51	1148.4634
	std	38.0069	480.7604	90.9351	72.0559	146.9477	77.732
F28	mean	8869.793	401.5093	1597.8946	6030.7752	498.645	400.0019
	std	510.7043	0.46578	1280.6178	994.7851	549.2319	0.00090197
UniNum	mean	0	2	0	0	3	0
	std	0	1	0	0	3	1
MulNum	mean	0	2	0	1	0	12
	std	4	1	1	1	0	8
ComNum	mean	0	2	0	0	0	6
	std	4	3	0	0	0	1
Total	mean	0	6	0	1	3	18
	std	8	5	1	1	3	10
Friedman's		5.7143	3.1071	3.6071	4.8571	2.25	1.4643

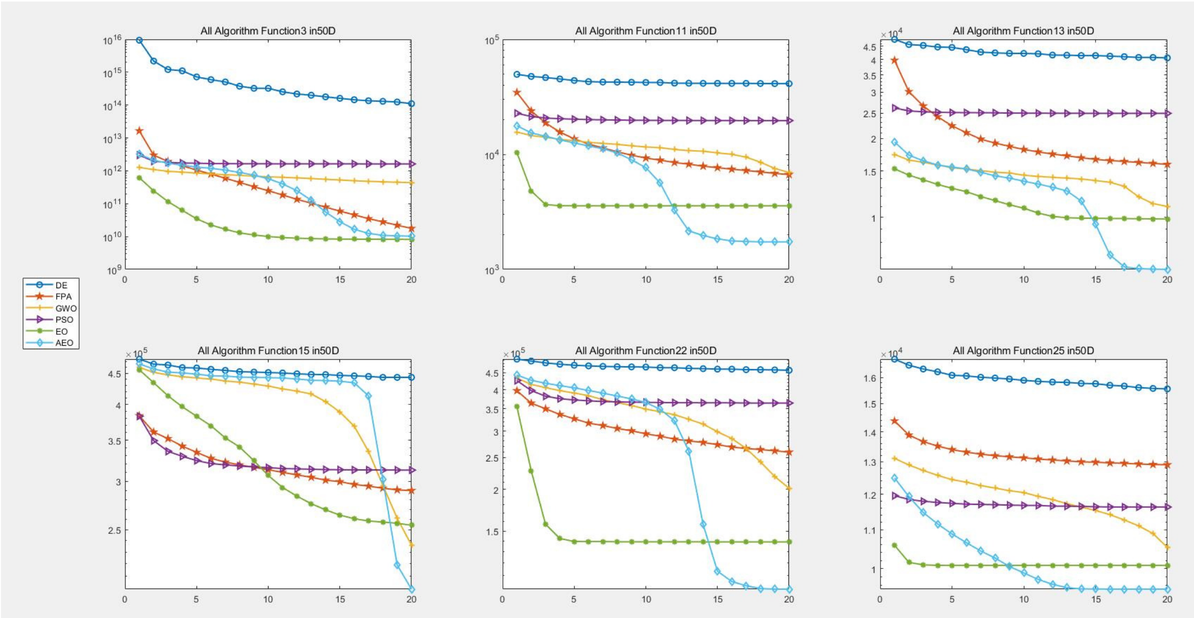

FIGURE 6. The convergence curve of AEO with the dimension of 50.

difficult and challenging problems than other compared algorithms. Moreover, from each table's Friedman's mean rank, AEO has pretty good results. Therefore, the convergence accuracy of AEO is better. In std item of Total, AEO also has a better performance than GWO, EO, PSO, DE, and FPA. Therefore, the convergence of AEO is quite stable.

From the figure 4 to 6, the convergence curves of AEO are basically at the bottom, which proves that the convergence accuracy is good. From the trend of the curve, AEO has a certain degree of decline with the iteration, that is, it can jump out of local optimal, which shows that AEO has a better global exploration ability.

5.3. **Performance test of two operators.** We also test the performance of the two operators. The fixed iterations are 2000, the algorithm is executed 31 times, and the dimension range is 10, 30, 50. Other settings are the same as above. In table 6 and 7, UN, MN, CN and To correspond to UniNum, MulNum, ComNum and Total.

As shown in table 6, whether in mean item or std item, QEO is better than EO, especially for multimodal and composed functions, proving that novel quantum operator effectively improves the EO performance, mainly exploration ability. The mean and std results reflect that the convergence accuracy and stability of QEO is better.

As can be seen from table 7, the global exploration ability of FEO is obviously stronger than that of EO, which shows that FPA-inspired pollination operator can also effectively improve the exploration ability, convergence accuracy and stability of EO.

5.4. **Performance test of multi-population method.** We also test the performance of multi-population method. The fixed iterations are 2000, the algorithm is executed 31 times, and the dimension range is 5, 10, 20. Other settings are the same as above. In table 8, UN, MN, CN and To correspond to UniNum, MulNum, ComNum and Total.

From table 8, in mean item and std item, MEO is better, especially in multimodal and composed functions, showing that multi-population method effectively improves the EO performance, mainly exploration ability. The results reflect that the convergence accuracy and stability of MEO is well.

TABLE 6. Experiment result of EO with novel quantum operator

Fun	D=10				D=30				D=50			
	EO		QEO		EO		QEO		EO		QEO	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
F1	7.3e-15	4.e-14	0.0e+0	0.0e+0	5.6e-13	1.6e-13	6.5e-13	2.1e-13	3.0e-12	2.4e-12	1.7e-9	4.3e-9
F2	5.61e+4	4.31e+4	7.48e+4	4.79e+4	2.26e+6	8.95e+5	2.98e+6	1.13e+6	4.47e+6	1.5e+6	9.7e+6	3.29e+6
F3	3.31e+5	8.21e+5	1.25e+5	2.28e+5	4.37e+7	5.78e+7	3.24e+7	3.97e+7	5.18e+8	5.12e+8	6.72e+8	3.46e+8
F4	5.35e+1	4.82e+1	8.29e+1	1.37e+2	3.75e+3	1.8e+3	3.59e+3	1.62e+3	1.31e+4	3.42e+3	1.47e+4	4.15e+3
F5	7.3e-14	6.3e-14	3.7e-14	5.4e-14	7.2e-13	2.4e-13	1.6e-12	2.1e-12	1.5e-10	3.5e-10	5.2e-06	7.1e-6
F6	7.01e+0	4.45e+0	8.32e+0	3.46e+0	3.41e+1	2.65e+1	2.41e+1	1.3e+1	7.02e+1	2.6e+1	5.31e+1	1.63e+1
F7	3.9e-1	6.98e-1	6.14e-1	1.9e+0	2.19e+1	1.27e+1	1.33e+1	7.27e+0	5.28e+1	1.42e+1	3.63e+1	1.02e+1
F8	2.03e+1	6.8e-2	2.03e+1	6.16e-2	2.1e+1	4.99e-2	2.1e+1	5.03e-2	2.12e+1	3.36e-2	2.12e+1	3.44e-2
F9	2.76e+0	1.08e+0	2.26e+0	1.43e+0	1.91e+1	4.51e+0	1.8e+1	4.03e+0	3.85e+1	8.77e+0	4.05e+1	5.88e+0
F10	1.33e-1	6.34e-2	1.38e-1	4.65e-2	1.38e-1	5.86e-2	9.29e-2	6.2e-2	5.66e-1	3.63e-1	1.7e+0	4.1e-1
F11	1.12e+0	1.11e+0	2.6e+0	1.49e+0	4.27e+1	1.32e+1	3.01e+1	9.61e+0	1.32e+2	2.86e+1	7.07e+1	1.38e+1
F12	1.1e+1	6.39e+0	6.52e+0	3.01e+0	6.82e+1	2.13e+1	4.47e+1	1.24e+1	1.5e+2	3.46e+1	1.06e+2	2.76e+1
F13	1.68e+1	6.93e+0	1.19e+1	7.88e+0	1.45e+2	2.84e+1	9.9e+1	2.56e+1	3.09e+2	5.56e+1	2.36e+2	3.49e+1
F14	1.83e+2	1.37e+2	8.6e+1	9.45e+1	2.0e+3	5.59e+2	2.11e+3	6.58e+2	4.35e+3	7.82e+2	3.65e+3	9.5e+2
F15	6.12e+2	2.49e+2	5.01e+2	2.2e+2	4.3e+3	6.92e+2	3.44e+3	7.19e+2	9.02e+3	1.26e+3	8.25e+3	9.93e+2
F16	5.2e-1	1.78e-1	4.49e-1	2.3e-1	1.29e+0	4.26e-1	1.37e+0	3.1e-1	2.18e+0	5.59e-1	2.35e+0	5.87e-1
F17	1.5e+1	2.63e+0	1.48e+1	2.23e+0	7.57e+1	1.44e+1	6.98e+1	1.34e+1	1.8e+2	3.49e+1	1.42e+2	2.17e+1
F18	1.93e+1	5.01e+0	1.57e+1	2.92e+0	1.24e+2	2.88e+1	9.52e+1	1.96e+1	2.77e+2	3.81e+1	2.21e+2	3.42e+1
F19	6.72e-1	1.76e-1	7.21e-1	2.03e-1	3.78e+0	1.1e+0	3.14e+0	6.03e-1	8.54e+0	1.92e+0	6.78e+0	1.43e+0
F20	2.3e+0	6.04e-1	2.26e+0	7.53e-1	1.1e+1	8.45e-1	1.05e+1	7.29e-1	1.98e+1	8.56e-1	1.97e+1	7.78e-1
F21	4.0e+2	2.8e-13	4.0e+2	2.9e-13	3.21e+2	7.06e+1	3.04e+2	7.36e+1	9.47e+2	1.41e+02	8.83e+2	3.00e+2
F22	2.57e+2	1.43e+2	2.33e+2	1.11e+2	1.79e+3	6.45e+2	1.93e+3	4.31e+2	4.51e+3	9.14e+2	4.30e+3	1.18e+3
F23	6.40e+2	2.20e+2	4.84e+2	1.85e+2	4.53e+3	5.58e+2	3.42e+3	6.49e+2	9.12e+3	1.17e+3	8.09e+3	1.02e+3
F24	2.12e+2	5.33e+0	2.12e+2	3.15e+0	2.46e+2	1.45e+1	2.48e+2	7.83e+0	2.93e+2	1.14e+1	2.93e+2	1.22e+1
F25	2.09e+2	5.24e+0	2.10e+2	4.40e+0	2.67e+2	1.01e+1	2.57e+2	7.36e+0	3.25e+2	1.42e+1	3.13e+2	9.43e+0
F26	1.73e+2	5.70e+1	1.99e+2	4.90e+1	2.82e+2	7.14e+1	2.39e+2	6.17e+1	3.77e+2	4.89e+1	3.72e+2	5.91e+1
F27	4.32e+2	1.05e+2	4.53e+2	1.01e+2	7.16e+2	1.01e+2	7.36e+2	8.65e+1	1.30e+3	1.79e+2	1.27e+3	1.27e+2
F28	3.44e+2	9.90e+1	2.99e+2	7.35e+1	2.87e+2	4.99e+1	3.35e+2	1.96e+2	4.00e+2	8.37e-8	8.00e+2	1.06e+3
UN	2.00e+0	2.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0	2.00e+0	2.00e+0	5.00e+0	4.00e+0	0.00e+0	1.00e+0
MN	6.00e+0	7.00e+0	9.00e+0	8.00e+0	3.00e+0	4.00e+0	1.20e+1	1.10e+1	4.00e+0	4.00e+0	1.10e+1	1.10e+1
CN	5.00e+0	1.00e+0	4.00e+0	7.00e+0	4.00e+0	3.00e+0	4.00e+0	5.00e+0	1.00e+0	5.00e+0	7.00e+0	3.00e+0
To	1.30e+1	1.00e+1	1.60e+1	1.80e+1	1.00e+1	1.00e+1	1.80e+1	1.80e+1	1.00e+1	1.30e+1	1.80e+1	1.50e+1

5.5. **Applying for EVRPTW.** In this section, we apply AEO for solving EVRPTW. We use instance from [44] to test performance of algorithm. Because there are few instances, we rotate, translate, and scale data of [44]. We rotate the data counterclockwise around the depot, 90 degrees, 180 degrees, 270 degrees. We shift the data in the direction and distance from [0,0] to [1,1], in the direction and distance from [0,0] to [1,-1], in the direction and distance from [0,0] to [-1,1], in the direction and distance from [0,0] to [-1,-1]. We take the depot as the center and scale the data by 0.25. 0.5, 0.75, 1.25, 1.5 times. On the whole, we test in 13 instances. For more effectively, we run 31 times for getting more reliable data. The particles are 100. The iteration is 500. The other parameters are the same as above mentioned. The EVRPTW experiment is shown in table 9.

In table 9, AEO, DE, FPA, GWO, PSO and EO are tested in 13 instances. The BKV item means the best known value of instance obtained by [44]. The best item is the best value of 31 runs, the mean item is the mean of 31 runs. The bestTotal item and meanTotal item are to calculate the performance of each algorithm in each instance. If an algorithm achieves the best result in an instance, then its corresponding term (bestTotal / meanTotal) is added to 1. Whether in bestTotal item or meanTotal item, AEO performs better than DE, FPA, GWO, PSO and EO. According to the Friedman's score, AEO performs best in these algorithms. On the whole, the AEO has better convergence accuracy in EVRPTW. Figure 7 is the best route result of the origin instance solved by AEO. It has three routes, and the its fitness is 6465.0939, is better than BKV [44]. It is proved that AEO can effectively solve EVRPTW.

TABLE 7. Experiment result of EO with FPA-inspired pollination operator

Fun	D=10				D=30				D=50			
	EO		FEO		EO		FEO		EO		FEO	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
F1	7.3e-15	4.1e-14	2.9e-14	7.8e-14	5.6e-13	1.6e-13	7.3e-13	1.8e-13	3.0e-12	2.4e-12	4.1e-12	1.5e-12
F2	5.61e+4	4.31e+4	4.45e+4	3.35e+4	2.26e+6	8.95e+5	2.37e+6	9.08e+5	4.47e+6	1.50e+6	5.06e+6	1.84e+6
F3	3.31e+5	8.21e+5	8.29e+4	2.75e+5	4.37e+7	5.78e+7	3.49e+7	7.90e+7	5.18e+8	5.12e+8	4.39e+8	4.61e+8
F4	5.35e+1	4.82e+1	4.85e+1	5.13e+1	3.75e+3	1.80e+3	2.77e+3	1.20e+3	1.31e+4	3.42e+3	1.12e+4	3.75e+3
F5	7.3e-14	6.3e-14	1.3e-13	6.6e-14	7.2e-13	2.4e-13	2.9e-12	1.3e-12	1.5e-10	3.5e-10	3.5e-08	4.5e-08
F6	7.01e+0	4.45e+0	9.19e+0	2.42e+0	3.41e+1	2.65e+1	2.96e+1	2.44e+1	7.02e+1	2.60e+1	6.68e+1	3.03e+1
F7	3.90e-1	6.98e-1	1.36e-1	1.79e-1	2.19e+1	1.27e+1	1.77e+1	1.21e+1	5.28e+1	1.42e+1	3.86e+1	7.19e+
F8	2.03e+1	6.80e-2	2.03e+1	6.34e-2	2.10e+1	4.99e-2	2.10e+1	4.24e-2	2.12e+1	3.36e-2	2.12e+1	4.27e-2
F9	2.76e+0	1.08e+0	2.75e+0	1.15e+0	1.91e+1	4.51e+0	1.70e+1	3.18e+0	3.85e+1	8.77e+0	3.33e+1	5.62e+0
F10	1.33e-1	6.34e-2	1.09e-1	5.25e-2	1.38e-1	5.86e-2	1.36e-1	5.90e-2	5.66e-1	3.63e-1	4.97e-1	3.73e-1
F11	1.12e+0	1.11e+0	9.31e-1	9.93e-1	4.27e+1	1.32e+1	2.90e+1	9.41e+0	1.32e+2	2.86e+1	9.61e+1	2.08e+1
F12	1.10e+1	6.39e+0	1.22e+1	5.08e+0	6.82e+1	2.13e+1	6.73e+1	2.07e+1	1.50e+2	3.46e+1	1.58e+2	3.13e+1
F13	1.68e+1	6.93e+0	1.88e+1	7.37e+0	1.45e+2	2.84e+1	1.41e+2	3.18e+1	3.09e+2	5.56e+1	2.81e+2	4.98e+1
F14	1.83e+2	1.37e+2	1.15e+2	1.24e+2	2.00e+3	5.59e+2	1.67e+3	4.32e+2	4.35e+3	7.82e+2	2.95e+3	6.72e+2
F15	6.12e+2	2.49e+2	5.22e+2	2.66e+2	4.30e+3	6.92e+2	3.51e+3	6.63e+2	9.02e+3	1.26e+3	7.19e+3	1.24e+3
F16	5.20e-1	1.78e-1	4.09e-1	1.44e-1	1.29e+0	4.26e-1	1.24e+0	3.90e-1	2.18e+0	5.59e-1	1.87e+0	4.87e-1
F17	1.50e+1	2.63e+0	1.57e+1	2.23e+0	7.57e+1	1.44e+1	8.12e+1	1.02e+1	1.80e+2	3.49e+1	1.65e+2	2.51e+1
F18	1.93e+1	5.01e+0	1.69e+1	3.27e+0	1.24e+2	2.88e+1	7.84e+1	1.33e+1	2.77e+2	3.81e+1	1.79e+2	2.95e+1
F19	6.72e-1	1.76e-1	6.64e-1	1.90e-1	3.78e+0	1.10e+0	3.34e+0	7.56e-1	8.54e+0	1.92e+0	8.48e+0	2.02e+0
F20	2.30e+0	6.04e-1	2.16e+0	5.13e-1	1.10e+1	8.45e-1	1.01e+1	8.11e-1	1.98e+1	8.56e-1	1.89e+1	8.76e-1
F21	4.00e+2	2.8e-13	4.00e+2	2.7e-13	3.21e+2	7.06e+1	3.28e+2	5.76e+1	9.47e+2	1.41e+2	9.66e+2	1.45e+2
F22	2.57e+2	1.43e+2	1.81e+2	1.20e+2	1.79e+3	6.45e+2	1.35e+3	4.91e+2	4.51e+3	9.14e+2	3.61e+3	9.87e+2
F23	6.40e+2	2.20e+2	6.61e+2	2.24e+2	4.53e+3	5.58e+2	3.52e+3	7.60e+2	9.12e+3	1.17e+3	7.45e+3	1.06e+3
F24	2.12e+2	5.33e+0	2.10e+2	4.77e+0	2.46e+2	1.45e+1	2.45e+2	1.34e+1	2.93e+2	1.14e+1	2.87e+2	1.34e+1
F25	2.09e+2	5.24e+0	2.08e+2	5.89e+0	2.67e+2	1.01e+1	2.62e+2	8.12e+0	3.25e+2	1.42e+1	3.20e+2	1.03e+1
F26	1.73e+2	5.70e+1	1.86e+2	5.78e+1	2.82e+2	7.14e+1	2.74e+2	6.84e+1	3.77e+2	4.89e+1	3.84e+2	1.07e+1
F27	4.32e+2	1.05e+2	4.23e+2	1.08e+2	7.16e+2	1.01e+2	6.96e+2	9.75e+1	1.30e+3	1.79e+2	1.20e+3	1.23e+2
F28	3.44e+2	9.90e+1	3.06e+2	2.50e+1	2.87e+2	4.99e+1	2.94e+2	3.59e+1	4.00e+2	8.37e-8	6.98e+2	9.27e+2
UN	2.00e+0	3.00e+0	3.00e+0	2.00e+0	3.00e+0	4.00e+0	2.00e+0	1.00e+0	3.00e+0	3.00e+0	2.00e+0	2.00e+0
MN	5.00e+0	4.00e+0	1.00e+1	1.10e+1	1.00e+0	2.00e+0	1.40e+1	1.30e+1	1.00e+0	5.00e+0	1.40e+1	1.00e+1
CN	3.00e+0	4.00e+0	6.00e+0	4.00e+0	2.00e+0	1.00e+0	6.00e+0	7.00e+0	3.00e+0	4.00e+0	5.00e+0	4.00e+0
To	1.00e+1	1.10e+1	1.90e+1	1.70e+1	6.00e+0	7.00e+0	2.20e+1	2.10e+1	7.00e+0	1.20e+1	2.10e+1	1.60e+1

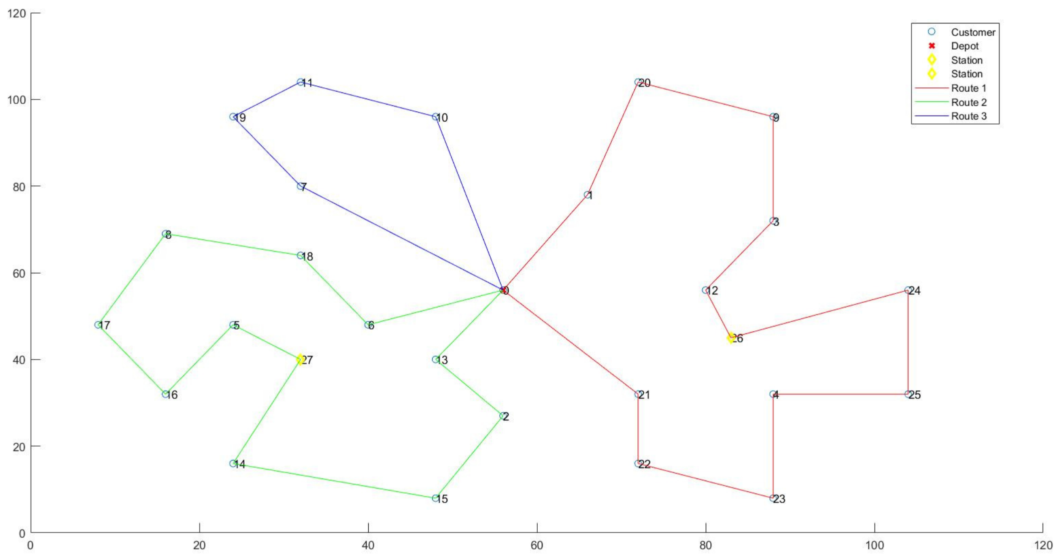

FIGURE 7. The best route result of the Origin Instance by AEO.

TABLE 8. Experiment result of EO with multi-population method

Fun	D=5				D=10				D=20			
	EO		MEO		EO		MEO		EO		MEO	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
F1	0	0	0	0	7.3e-15	4.1e-14	0.00e+00	0.00e+0	2.2e-13	1.1e-13	4.4e-13	1.7e-13
F2	2.20e+3	2.79e+3	4.27e+3	4.13e+3	5.61e+4	4.31e+4	6.67e+4	4.07e+4	5.68e+5	2.59e+5	9.42e+5	3.70e+5
F3	6.48e+0	3.15e+0	1.89e+0	2.02e+0	3.31e+5	8.21e+5	4.41e+4	1.25e+5	3.68e+6	1.35e+7	1.86e+6	2.62e+6
F4	1.14e-2	1.64e-2	1.24e+0	1.11e+0	5.35e+1	4.82e+1	8.92e+2	6.16e+2	7.77e+2	3.14e+2	6.92e+3	2.33e+3
F5	0	0	0	0	7.3e-14	6.3e-14	9.2e-14	4.6e-14	2.5e-13	9.9e-14	9.7e-13	2.8e-13
F6	1.32e-2	8.50e-3	3.25e-2	2.47e-2	7.01e+0	4.45e+0	3.85e+0	4.82e+0	3.35e+0	1.21e+1	4.02e-1	6.69e-1
F7	2.84e-2	6.61e-2	1.30e-2	3.27e-2	3.9e-01	6.98e-1	9.62e-1	1.30e+0	3.90e+0	3.44e+0	5.85e+0	3.79e+0
F8	1.08e+1	9.76e+0	3.75e+0	6.97e+0	2.03e+1	6.8e-02	2.03e+1	7.15e-2	2.08e+1	5.23e-2	2.08e+1	5.76e-2
F9	5.36e-1	6.22e-1	2.40e-1	3.35e-1	2.76e+0	1.08e+0	2.39e+0	7.85e-1	9.46e+0	3.00e+0	9.01e+0	1.67e+0
F10	6.54e-2	4.75e-2	5.07e-2	2.10e-2	1.33e-1	6.34e-2	1.42e-1	6.88e-2	1.14e-1	5.85e-2	1.88e-1	7.24e-2
F11	3.21e-2	1.79e-1	0.00e+0	0.00e+0	1.12e+0	1.11e+0	2.18e+0	1.13e+0	1.66e+1	6.80e+0	1.98e+1	5.42e+0
F12	1.60e+0	9.84e-1	1.83e+0	7.75e-1	1.10e+1	6.39e+0	7.89e+0	3.34e+0	2.77e+1	8.57e+0	2.65e+1	7.87e+0
F13	3.19e+0	2.34e+0	2.10e+0	1.25e+0	1.68e+1	6.93e+0	1.39e+1	5.17e+0	6.13e+1	2.32e+1	5.99e+1	1.57e+1
F14	1.80e+1	2.38e+1	4.85e+0	7.36e+0	1.83e+2	1.37e+2	1.08e+2	8.00e+1	1.12e+3	3.19e+2	9.20e+2	2.44e+2
F15	6.88e+1	7.39e+1	2.28e+1	1.71e+1	6.12e+2	2.49e+2	4.09e+2	2.05e+2	2.14e+3	4.42e+2	1.86e+3	4.75e+2
F16	2.33e-1	1.45e-1	1.66e-1	7.37e-2	5.20e-1	1.78e-1	4.03e-1	1.21e-1	9.08e-1	2.95e-1	8.67e-1	2.26e-1
F17	5.36e+0	2.69e-1	5.24e+0	7.92e-1	1.50e+1	2.63e+0	1.52e+1	2.25e+0	4.28e+1	7.71e+0	4.15e+1	5.03e+0
F18	6.03e+0	1.39e+0	5.68e+0	1.23e+0	1.93e+1	5.00e+0	1.96e+1	2.40e+0	5.70e+1	1.10e+1	6.74e+1	8.40e+0
F19	1.42e-1	7.38e-2	1.28e-1	7.37e-2	6.72e-1	1.76e-1	6.32e-1	1.39e-1	2.05e+0	8.27e-1	1.85e+0	3.23e-1
F20	4.85e-1	4.43e-1	9.61e-2	6.91e-2	2.30e+0	6.04e-1	1.88e+0	2.94e-1	5.77e+0	9.45e-1	5.99e+0	7.88e-1
F21	2.55e+2	8.50e+1	2.10e+2	1.04e+2	4.00e+2	2.8e-13	4.00e+2	2.9e-13	3.65e+2	7.09e+1	2.65e+2	8.39e+1
F22	1.69e+2	1.24e+2	8.32e+1	6.49e+1	2.57e+2	1.43e+2	1.50e+2	1.01e+2	1.20e+3	4.07e+2	1.05e+3	2.12e+2
F23	2.34e+2	9.68e+1	1.42e+2	6.61e+1	6.40e+2	2.20e+2	5.51e+2	2.00e+2	2.39e+3	5.20e+2	2.06e+3	4.74e+2
F24	1.12e+2	3.04e+1	4.00e+1	3.24e+1	2.12e+2	5.33e+0	2.04e+2	1.42e+1	2.32e+2	5.30e+0	2.31e+2	4.61e+0
F25	1.02e+2	2.72e+0	1.01e+2	8.77e+0	2.09e+2	5.24e+0	2.05e+2	4.39e+0	2.40e+2	7.49e+0	2.38e+2	4.00e+0
F26	1.02e+2	1.32e+0	5.27e+1	4.63e+1	1.73e+2	5.70e+1	1.32e+2	3.80e+1	2.08e+2	3.13e+1	2.00e+2	4.98e-2
F27	3.13e+2	2.03e+1	3.04e+2	9.99e+0	4.32e+2	1.05e+2	3.17e+2	4.21e+1	5.56e+2	8.59e+1	5.43e+2	6.16e+1
F28	2.97e+2	4.07e+1	2.68e+2	7.48e+1	3.44e+2	9.90e+1	3.03e+2	1.80e+1	8.92e+2	4.47e+2	2.81e+2	6.01e+1
UN	4.00e+0	4.00e+0	3.00e+0	3.00e+0	3.00e+0	1.00e+0	2.00e+0	4.00e+0	4.00e+0	4.00e+0	1.00e+0	1.00e+0
MN	2.00e+0	2.00e+0	1.30e+1	1.30e+1	6.00e+0	5.00e+0	9.00e+0	1.00e+1	5.00e+0	4.00e+0	1.00e+1	1.10e+1
CN	0.00e+0	5.00e+0	8.00e+0	3.00e+0	1.00e+0	2.00e+0	8.00e+0	6.00e+0	0.00e+0	1.00e+0	8.00e+0	7.00e+0
To	6.00e+0	1.10e+1	2.40e+1	1.90e+1	1.00e+1	8.00e+0	1.90e+1	2.00e+1	9.00e+0	9.00e+0	1.90e+1	1.90e+1

6. Conclusions.

We propose AEO, which improves EO with the multi-population method, novel quantum operator, and FPA-inspired pollination operator, in this work. The multi-population method is the basic framework of the AEO. The novel quantum operator effectively enhances EO's global exploration capability. FPA-inspired pollination operator brings the Levy flight of FPA into EO, which also effectively improves the global exploration capability of the EO. Then we test the AEO by CEC2013. AEO has better convergence performance than DE, FPA, GWO, PSO, and EO. We test two operators, and the results show the operators' good effect on global exploration. We also test multi-population method that improvs EO's convergence arrcuracy and stability, as is shown in results. Finally, AEO is applied to solve EVRPTW. From the test results of instances, AEO is more powerful to solve EVRPTW than DE, FPA, GWO, PSO, and EO.

In the future, the AEO could be further improved, such as hybrid [47], and adding chaotic mapping [48]. AEO proposed in this paper could also be applied to other fields, such as power system problems [49], wireless sensor networks problems [50], ontology matching domain [51] and smart city traffic network prediction [52].

Acknowledgment. This work is supported by the National Natural Science Foundation of China (61872085) and the Natural Science Foundation of Fujian Province (2018J01638).

TABLE 9. EVRPTW result of algorithms in Instance

Instance	BKV	item	DE	FPA	GWO	PSO	EO	AEO
Origin		best	11721.7505	9010.2137	7097.3954	7744.5586	7448.2776	6465.0939
		mean	13328.5274	9837.5337	8289.2483	9621.0768	8595.2492	7838.2247
Rotate-90		best	12610.6567	8796.8262	6959.2419	8295.0263	7057.2635	7132.8642
		mean	13426.8159	9662.4003	8637.2758	9625.7495	8494.3244	8041.1428
Rotate-180		best	12519.9547	8978.0223	7044.3998	8695.7266	6729.9164	6670.298
		mean	13528.0262	9810.7743	8346.0318	10057.749	8393.1087	7784.6456
Rotate-270	6542.7	best	11717.0043	8457.6427	7087.1	8632.441	7318.731	6694.1434
		mean	13407.8851	9667.9443	8344.6336	9969.0214	8487.0366	7710.7842
Shift from [0,0]		best	12516.7298	8934.3689	7116.4766	8225.2407	6905.7139	6564.8213
to [1,1]		mean	13509.6714	9645.0212	8251.4028	9761.1078	8401.2048	7866.4214
Shift from [0,0]		best	12174.1286	8537.256	6554.7396	8049.4606	6710.1168	6638.3542
to [1,-1]		mean	13393.4058	9731.0928	8349.9094	9788.338	8517.8441	7929.9351
Shift from [0,0]		best	12453.1163	8344.3826	7432.9681	7834.4922	7581.5427	6841.1297
to [-1,1]		mean	13438.4318	9719.2028	8636.0574	9720.6479	8486.0977	7905.2889
Shift from [0,0]		best	12508.1914	8265.1345	7059.7453	8213.9809	7253.0923	6362.046
to [-1,-1]		mean	13438.391	9590.5688	8562.5893	9706.0624	8278.6911	7947.7014
Scale 0.25 times		best	3098.6277	2374.2175	1885.638	2010.0587	1797.6263	1831.5206
		mean	3313.9794	2479.9336	2167.2615	2703.3838	2224.9164	2049.8292
Scale 0.5 times		best	6007.8215	4008.9649	3598.1882	4043.5139	3343.6505	3406.9459
		mean	6348.0743	4726.671	4196.4177	4882.121	4031.1932	3950.5976
Scale 0.75 times		best	8496.6926	6620.8064	5271.673	5795.863	5094.6056	4800.0002
		mean	9571.8519	7078.4849	6367.9065	7274.6251	6245.7622	5670.2391
Scale 1.25 times		best	16836.0697	11532.0645	9422.0926	10224.0463	9456.0459	9102.1601
		mean	83305.763	12811.8145	11279.1559	12015.0672	10983.8487	10578.1064
Scale 1.75 times		best	19771.4803	13800.7503	11732.3469	13231.066	11908.2651	11346.6621
		mean	1895217.572	16359.2715	13854.685	854846.2115	13762.1787	13237.0302
bestTotal			0	0	2	0	2	9
meanTotal			0	0	0	0	0	13
Friedman's			6	4.2308	2.5385	4.7692	2.4615	1

REFERENCES

[1] H. Ceylan, H. Ceylan, S. Haldenbilen and O. Baskan, Transport energy modeling with meta-heuristic harmony search algorithm, an application to Turkey, *Energy Policy*, vol. 36, no. 7, pp. 2527–2535, 2008.

[2] Z. Beheshti and S. M. H. Shamsuddin, A review of population-based meta-heuristic algorithms, *International Journal of Advances in Soft Computing and its Applications*, vol. 5, no. 1, pp. 1–35, 2013.

[3] R. Storn and K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[4] Z. Meng, J. S. Pan and L. Kong, Parameters with adaptive learning mechanism (PALM) for the enhancement of differential evolution, *Knowledge-Based Systems*, vol. 141, pp. 92–112, 2018.

[5] X. S. Yang and S. Deb, Cuckoo search via Lévy flights, *2009 World Congress on Nature and Biologically Inspired Computing (NaBIC)*, Coimbatore, India, pp. 210–214, 2009.

[6] J. S. Pan, P. C. Song, S. C. Chu and Y. J. Peng, Improved Compact Cuckoo Search Algorithm Applied to Location of Drone Logistics Hub, *Mathematics*, vol. 8, no. 3, pp. 333, 2020.

[7] J. Kennedy and R. C. Eberhart, Particle Swarm Optimization, *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Western Australia, pp. 1942–1948, 1995.

[8] K. Wang, Y. Liu, D. Liu and Q. Shi, Improved particle swarm optimization algorithm based on gaussian-grid search method, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 9, no. 4, pp. 1031–1037, 2018.

[9] D. Karaboga and B. Basturk, Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems, *International Fuzzy Systems Association World Congress*, Cancun, Mexico, pp. 789–798, 2007.

[10] C. M. Wu, Z. Y. Dong and X. Yang, An Improved Multidimensional Calibration Algorithm Based on Artificial Bee Colony Algorithm, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 9, no. 5, pp. 1080–1086, 2018.

[11] X. S. Yang, Flower Pollination Algorithm for Global Optimization, *International Conference on Unconventional Computing and Natural Computation*, Orléans, France, pp. 240–249, 2012.

[12] X. S. Yang, *Nature-Inspired Optimization Algorithms*, Elsevier, 2014.

[13] S. Mirjalili, How effective is the Grey Wolf optimizer in training multi-layer perceptrons, *Applied Intelligence*, vol. 43, no. 1, pp. 150–161, 2015.

[14] J. S. Pan, P. Hu, S. C. Chu, Novel parallel heterogeneous meta-heuristic and its communication strategies for the prediction of wind power, *Processes*, vol. 7, no. 11, 2019.

[15] Z. Meng, J. S. Pan and H. Xu, QUasi-Affine TRansformation Evolutionary (QUATRE) algorithm: A cooperative swarm based algorithm for global optimization, *Knowledge-Based Systems*, vol. 109, pp. 104–121, 2016.

[16] N. Liu, J.-S. Pan, J. Wang and T.-T. Nguyen, An adaptation multi-group quasi-affine transformation evolutionary algorithm for global optimization and its application in node localization in wireless sensor networks, *Sensors*, vol. 19, no. 19, 2019.

[17] A. Faramarzi, M. Heidarinejad, B. Stephens and S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, *Knowledge-Based Systems*, vol. 191, 2020.

[18] J. S. Pan, P. W. Tsai and Y. B. Liao, Fish migration optimization based on the fishy biology, *2010 Fourth International Conference on Genetic and Evolutionary Computing*, Shenzhen, China, pp. 783–786, 2010.

[19] B. Guo, Z. Zhuang, J. S. Pan, S. C. Chu, Optimal design and simulation for PID controller using Fractional-Order Fish Migration Optimization algorithm, *IEEE Access*, vol. 9, pp. 8808–8819, 2021.

[20] Q. W. Chai, S. C. Chu, J. S. Pan and W. M. Zheng, Applying Adaptive and Self Assessment Fish Migration Optimization on Localization of Wireless Sensor Network on 3-D Terrain, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 11, no. 2, pp. 90–102, 2020.

[21] A. Wunnava, M. K. Naik, R. Panda, B. jena and A. Abraham, A novel interdependence based multilevel thresholding technique using adaptive equilibrium optimizer, *Engineering Applications of Artificial Intelligence*, vol. 94, 103836, 2020.

[22] Y. Gao, Y. Zhou, Q. Luo, An efficient binary equilibrium optimizer algorithm for feature selection, *IEEE Access*, vol. 8, pp. 140936–140963, 2020.

[23] D. T. Abdul-hamied, A. M. Shaheen, W. A. Salem, W. I. Gabr and R. A. El-sehiemy, Equilibrium optimizer based multi dimensions operation of hybrid AC/DC grids, *Alexandria Engineering Journal*, vol. 59, no. 6, pp. 4787–4803, 2020.

[24] A. Rabehi, B. Nail, H. Helal, A. Douara, A. Ziane, M. Amrani, B. Akkal and Z. Benamara, Optimal estimation of Schottky diode parameters using a novel optimization algorithm: Equilibrium optimizer, *Superlattices and Microstructures*, vol. 146, 106665, 2020.

[25] K. Nusair, L. Alhmoud, Application of Equilibrium Optimizer Algorithm for Optimal Power Flow with High Penetration of Renewable Energy, *Energies*, vol. 13, no. 22, 6066, 2020.

[26] J. F. Chang, S. C. Chu, J. F. Roddick and J. S. Pan, A parallel particle swarm optimization algorithm with communication strategies, *Journal of Information Science and Engineering*, vol. 21, no. 4, pp. 809–818, 2005.

[27] S. C. Chu, J. F. Roddick and J. S. Pan, Ant colony system with communication strategies, *Information Sciences*, vol. 167, no. 1-4, pp. 63–76, 2004.

[28] S. C. Chu, P. W. Tsai and J. S. Pan, Cat swarm optimization, *Pacific Rim International Conference on Artificial Intelligence*, Guilin, China, pp. 854–858, 2006.

[29] V. Snasel, L. Kong, P. W. Tsai and J. S. Pan, Sink Node Placement Strategies based on Cat Swarm Optimization Algorithm, *Journal of Network Intelligence*, vol. 1, no. 2, pp. 52–60, 2016.

[30] P. W. Tsai, J. S. Pan, S. M. Chen, B. Y. Liao and S. P. Hao, Parallel cat swarm optimization, *International Conference on Machine Learning and Cybernetics*, Kunming, China, pp. 3328–3333, 2008.

[31] P. W. Tsai, J. S. Pan, S.-M. Chen and B. Y. Liao, Enhanced parallel cat swarm optimization based on the Taguchi method, *Expert Systems with Applications*, vol. 39, no. 7, pp. 6309–6319, 2012.

[32] L. Kong, J. S. Pan, P. W. Tsai, S. Vaclav and J. H. Ho, A Balanced Power Consumption Algorithm Based on Enhanced Parallel Cat Swarm Optimization for Wireless Sensor Network, *International Journal of Distributed Sensor Networks*, vol. 11, no. 3, 729680, 2015.

[33] J. Sun, B. Feng and W. Xu, Particle swarm optimization with particles having quantum behavior, *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, Portland, USA, pp. 325–331, 2004.

[34] J. Sun, W. Fang, V. Palade, X. Wu and W. Xu, Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point, *Applied Mathematics and Computation*, vol. 218, no. 7, pp. 3763–3775, 2011.

[35] K. Lu and H. Li, Quantum-behaved flower pollination algorithm, *International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, Guiyang, China, pp. 66–69, 2015.

[36] G. B. Dantzig and J. H. Ramser, The truck dispatching problem, *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.

[37] Y. Zhou, L. Chen, Y. Yang, Y. Li, G. Cheng, Y. Fu, C. Zhu, Y. Liu and H. Mao, Electric vehicle routing problem: Model and algorithm, *International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Phuket, Thailand, pp. 1049–1054, 2020.

[38] Q. Wang, S. Peng and S. Liu, Optimization of Electric Vehicle Routing Problem Using Tabu Search, *Chinese Control And Decision Conference (CCDC)*, Hefei, China, pp. 2220–2224, 2020.

[39] J. K. Lenstra and A. H. G. R. Kan, Complexity of vehicle routing and scheduling problems, *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[40] S. Erdoğan and E. Miller-Hooks, A green vehicle routing problem, *Transportation research part E: Logistics and Transportation Review*, vol. 48, no. 1, pp. 100–114, 2012.

[41] J. Lin, W. Zhou and O. Wolfson, Electric vehicle routing problem, *Transportation Research Procedia*, vol. 12, pp. 508–521, 2016.

[42] M. Schneider, A. Stenger and D. Goeke, The electric vehicle-routing problem with time windows and recharging stations, *Transportation Science*, vol. 48, no. 4, pp. 500–520, 2014.

[43] S. R. Kancharla and G. Ramadurai, Electric vehicle routing problem with non-linear charging and load-dependent discharging, *Expert Systems with Applications*, vol. 160, 113714, 2020.

[44] S. Gao, *The Research of Vehicle Routing Problems with Time Window Based on Electric Vehicle*, M. Eng. Thesis, Dalian Maritime University, Dalian, China, 2015.

[45] J. J. Liang, B. Y. Qu, P. N. Suganthan and A. G. Hernández-Díaz, *Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimizatio*, Technical Report, 2013.

[46] W. Y. Szeto, Y. Wu and S. C. Ho, An artificial bee colony algorithm for the capacitated vehicle routing problem, *European Journal of Operational Research*, vol.215, no. 1, pp. 126–135, 2011.

[47] O. Abdel-Raouf, I. El-Henawy and M. Abdel-Baset, A novel hybrid flower pollination algorithm with chaotic harmony search for solving sudoku puzzles, *International Journal of Modern Education and Computer Science*, vol. 6, no. 3, pp. 38–44, 2014.

[48] H. T. T. Binh, N. T. Hanh and N. Dey, Improved cuckoo search and chaotic flower pollination optimization algorithm for maximizing area coverage in wireless sensor networks, *Neural Computing and Applications*, vol. 30, no. 7, pp. 2305–2317, 2018.

[49] H. M. Dubey, M. Pandit and B. K. Panigrahi, A biologically inspired modified flower pollination algorithm for solving economic dispatch problems in modern power systems, *Cognitive Computation*, vol. 7, no. 5, pp. 594–608, 2015.

[50] T. T. Nguyen, J. S. Pan and T. K. Dao, An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network, *IEEE Access*, vol. 7, pp. 75985–75998, 2019.

[51] X. Kou and J. Feng, Matching Ontologies through Compact Monarch Butterfly Algorithm, *Journal of Network Intelligence*, vol. 5, no. 4, pp. 191–197, 2020.

[52] F. Zhang, T. Y. Wu, Y. Wang, R. Xiong, G. Ding, P. Mei, L. Liu Application of quantum genetic optimization of LVQ neural network in smart city traffic network prediction, *IEEE Access*, vol. 8, pp. 104555–104564, 2020.