An Improved Dragonfly Algorithm With Higher Exploitation Capability to Optimize the Design of Hybrid Power Active Filter

WANXUAN DAI, CHUNQUAN LI, (Member, IEEE), ZHILING CUI, YUFAN WU, LEYINGYUE ZHANG, AND JUNRU HUANG
School of Information Engineering, Nanchang University, Nanchang 330029, China

ABSTRACT
Hybrid power active filter (HAPF) is an important device to suppress the harmonics of the power system. In HAPF, the parameters estimation has a great impact on ensuring the power quality in the power system. Aiming at the problem of minimizing the harmonic pollution (HP) in the power system, this paper proposes a new technology namely IEDA for parameter optimization of hybrid power active filters, which is an improved dragonfly algorithm (DA) with higher exploitation capability. DA is a global search algorithm with sufficient ability to avoid falling into local optimization, however, DA performs poorly for local search. In the IEDA, we adopt a strategy of division of labor to divide particles into exploitation population and exploration population. In the exploitation population, we introduce the information exchange mechanism of the differential evolution (DE) and set up an exemplar pool to enhance its exploitation capability. In the exploration population, we use the global search ability of the DA to prevent particles from falling into a local optimum. Through the division of labor between exploitation population and exploration population, the problems of low accuracy and slow convergence of DA are effectively solved. Experimental results show that the algorithm has greatly improved accuracy and reliability compared with seven well-established algorithms.

INDEX TERMS
Hybrid active power filter (HAPF), harmonic pollution (HP), dragonfly algorithm (DA), global optimization.

I. INTRODUCTION
With the development of society and technology, non-linear shock loads have been continuously connected to power systems. Non-linear loads will generate two typical sources of harmonic pollution: harmonic current sources and harmonic voltage sources. Under the action of harmonics, the capacitor will generate additional power loss, the power factor (PF) of the load will decrease, and the power system loss will continue to increase [1]. In particular, due to the heavy use of sensitive equipment and instruments, power users have increasingly demanding power quality [2].

At present, the main measure for suppressing harmonics in the power grid is to install filtering devices. Commonly used filtering devices include passive power filter (PPF), active power filter (APF) and hybrid active filter (HAPF). The passive filters used in traditional harmonics and reactive power compensation have the advantages of simple structure, less equipment investment, and low operating costs, so they are widely used in power systems. However, there are also limitations in terms of higher harmonics and low filtering efficiency. Especially, when the harmonic distortion changes dynamically, it is not easy to compensate the harmonics [3]. APF is a new type of power electronic filtering equipment with fast response speed and flexible control. It can dynamically compensate harmonics whose size and frequency change, and overcome the shortcomings of PPF.
However, because it must withstand the line frequency working voltage, the VA rating of the power electronic converter in APF becomes very large, and the huge VA converter rated power will bring high cost, high electromagnetic interference and high power loss [4]. Besides, APF is limited by switching devices and its relatively small capacity also restricts its application. Since the 1990s, a series of HAPF topologies have been proposed [5]–[7]. Hybrid active power filters combine the advantages of APF and PPF, break through the limitations of APF and PPF as well as solve the problems of high cost and small capacity of APF.

Real parameter global optimization plays an important role in various scientific problems and engineering applications. The convex optimization problem can be solved by a classic gradient algorithm, such as gradient descent [8]. However, for multi-modal, discontinuous, non-convex and non-differentiable optimization problems, these gradient algorithms have encountered huge challenges. As a competitive choice, the meta-heuristic algorithm is not only insensitive to initial conditions, but also does not require the continuity of the solution space. Compared with traditional gradient-based optimization methods, these methods can be used to solve continuous, multi-modal, non-convex, multi-dimensional optimization problems with large differences in properties, such as [9], [10]. Most of these algorithms imitate human and natural behavior; they will automatically start to find the optimal solution after initialization, and do not involve complicated calculations. Therefore, the algorithm based on meta-heuristic is easy to implement, and the calculation burden is low. In recent decades, people have proposed meta-heuristic algorithms for global optimization, including Monarch Butterfly Optimization (MBO) [11], Moth Search Algorithm (MSA) [12], Sine Cosine Algorithm (SCA) [13], Moth-flame Optimization (MFO) [14], Particle Swarm Optimization (PSO) [15], Differential Evolution (DE) [16], Grasshopper Optimisation Algorithm (GOA) [17]. Moreover, many scholars have studied and proposed some variants to improve the performance of these meta-heuristic algorithms, such as Chaotic Krill Herd Algorithm (CKH) [18], enhanced fitness-adaptive differential evolution algorithm with novel mutation (EFADE) [19], Double Evolutionary Learning Moth-Flame Optimization (DELMFO) [20], Lévy-flight moth-flame optimization (LMFO) algorithm [21], a hybrid optimization technique combining genetic and exchange market algorithms (EMGA) [22], Chaos enhanced grey wolf optimization wrapped (ELM) [23]. These meta-heuristic algorithms are widely used to solve various practical engineering problems. For example, the parameter evaluation problem of photovoltaic model [24], the demand estimation of water resources [25], the flow shop scheduling [26] and big data optimization problems [27], [28]. Accordingly, these meta-heuristic techniques are also suitable for designing filters in power systems. The design of HAPF mainly depends on the design of PPF and APF. Various optimization algorithms such as particle swarm optimization (PSO) [29], differential evolution (DE) [30], genetic algorithm (GA) [31] and bat algorithm (BA) [32] have been applied to the design of PPF. However, the design complexity of HAPF is higher than that of PPF, and the optimal gain $K$ is generally predefined or determined by trial and error [33], so there is less literature on the application of meta-heuristic optimization techniques in HAPF design. Moreover, with the increase of non-linear loads, the complexity will continue to increase when there are also harmonics in the power supply. Tiwari et al. presents an efficient technique for harmonic compensation using Ant Colony Optimization (ACO) algorithm based on hybrid active power filter [34]. Jhapte et al. studied an optimal controller optimized by the harmony search algorithm (HSA) to minimize the total harmonics of the shunt hybrid active power filter [35]. And the literature [36], [37] and [38] proposed the design technique of HAPF, but did not increase the possibility of source nonlinearity. And under the premise of source nonlinearity, Biswas et al. proposed two popular topologies of hybrid power active filters and used the L-SHADE algorithm to estimate the HAPF parameters [39]. However, the L-SHADE algorithm has a low success rate and poor robustness. Therefore, it is necessary to develop a novel algorithm for parameter optimization of hybrid power active filters. Based on the two HAPF topologies proposed in [39], we propose an improved DA to estimate the parameters of HAPF.

DA is a new type of meta-heuristic algorithm proposed by Mirjalili in 2015 [40]. The DA is inspired by two behaviors of the dragonfly: hunting and migration, which correspond to the exploitation and exploration of meta-heuristic algorithm. The DA divides these two behaviors into five parts. Hunting behavior corresponds to enemy distraction and food attraction, while migration behavior corresponds to separation, alignment and cohesion. In the algorithm, when there are no other dragonflies near the current dragonfly, the dragonfly performs Lévy-Flight; when there are dragonflies near the current dragonfly, the dragonfly updates through these two behaviors. As the number of iterations increases, the algorithm dynamically controls these two behaviors to balance the exploitation ability and exploration ability of the dragonfly. By testing the standard benchmark function, the superior performance of the DA is demonstrated. Therefore, DA has received extensive attention since it was proposed and has been applied to many practical engineering problems. Hammouri et al. used the DA to optimize the traveling salesman problem based on data mining [41]. Compared with other algorithms, the DA has the shortest running time and performs satisfactory excellent. Lagos et al. used the dragonfly algorithm to solve the cable-stayed bridge reinforcement problem and achieved preeminent results [42]. In order to obtain better performance, a series of DA variant algorithms are proposed. Peng et al. introduced a novel chaotic dragonfly algorithm based on sine-cosine mechanism (SC-DA) for optimization design [43], Song et al. also proposed an elite opposition learning and exponential function steps-based dragonfly algorithm for global optimization [44]. And through numerical benchmark functions, the excellent performance of the two methods is verified. Ranjini et al.
presented a memory based hybrid dragonfly algorithm (MHDA) for three engineering design problems [45]. The difficulties of designing these three engineering problems were overcome successfully, which indicated the superiority of proposed algorithm. In addition, by testing the basic unconstrained benchmark function and CEC 2014 test function, the efficiency of MHDA was verified again. Shilaja et al. presented a hybrid algorithm based on enhance grey wolf optimization and algorithm of dragonfly algorithm (DA-EGWO) for handling OPF (optimal power flow) issues [46]. The presented method is investigated and experimented on IEEE 30 bus system. The hybrid EGWO–DA algorithm result proves that this method is more efficient in cost reduction of the IEEE 30 bus system. The hybrid EGWO–DA algorithm result proves that this method is more efficient in cost reduction and power loss minimization. Xu et al. proposed a color image segmentation method (IDA) based on DA and DE then used 8 color images in the Berkeley database to prove the superiority of the method [47].

According to the theory of no free lunch, no algorithm is universal in all cases [48], and many algorithms are not competitive in HAPF parameters estimation. Therefore, our work is to design an improved DA to accurately and reliably estimate the parameters of HAPF. In DA, the positions of dragonflies in the search space are randomly distributed. When there are no other particles around the current particle, these particles will execute the Lévy-Flight strategy. When there are other particles around the current particle, the migration behavior of the dragonfly will be dynamically changed to keep the dragonfly at a certain distance, which makes DA have excellent global search ability. However, in the DA, the previous generation of superior particles did not communicate with the next generation of particles, and the superior particles could not guide the population update, resulting in insufficient exploitation capability and impaired exploitation accuracy. Therefore, the poor performance of the DA algorithm is mainly due to the lack of an effective information feedback mechanism. Through the effective information feedback mechanism, the overall performance of the algorithm can be effectively improved [49], and some literature have proved that the introduction of the information feedback mechanism is effective [50], [51]. In order to overcome the shortcomings, we propose a new technology IEDA, which is an improved dragonfly algorithm with higher exploitation ability. The IEDA uses a division of labor strategy to divide particles into exploitation and exploitation populations. In addition, inspired by Ren et al. [52], we established an exemplar pool to store high-quality particle information. Different from [52], we store part of the individual optimal value instead of part of the global optimal value in the exemplar pool, so that the exploitation population can effectively use the information in the exemplar pool to jump out of the local optimum. We use the exemplar pool to connect the two populations and feedback high-quality particle information to the exploitation population. This forms an effective information feedback mechanism and improves the performance of the algorithm. The exploitation population introduces the information exchange mechanism “DE / current-to-best / 1” of the differential evolution. The random particles of the exploitation population can be selected from the exemplar pool or the exploitation population. The information exchange with high-quality particles enhances the population exploitation ability. However, since the quality of the particles in exemplar pool is not high in the early stage of the algorithm iteration, it is impossible to provide high-quality particle information to the exploitation population. Inspired by Liang et al. [53], we introduced a probability curve to control the selection of particle learning objects. This probability curve controls the exploitation ability of the exploitation population in the whole iteration process, so that the exploitation population has a certain exploration ability in the early iteration process. The exploration of the population uses the global search ability of the DA and has a strong ability to avoid falling into a local optimum. The exploration particles store the searched high-quality particles in the exemplar pool, which can prevent the exploitation population from falling into a local optimum. Through the complementary combination of these two populations, the exploitation performance has been improved very well while the exploration ability of the DA is well retained. In this model, compared with other well-known algorithms, the robustness and reliability of the IEDA are better than other algorithms.

The main contributions of this paper are as follows:
1) The exemplar pool strategy is introduced to provide high-quality particle information for exploitation population.
2) We propose a division of labor strategy, introduce the information exchange mechanism of DE into IEDA, and use the high-quality particles found by DA to guide the population update.
3) The proposed IDEA is applied to the parameter estimation of HAPF in power systems and compared with other mature algorithms. The experimental results show the effectiveness of the algorithm.

The rest of this article is as follows. Section II illustrates two HAPF models with the objective function and the study cases. Section III introduces the original DA technology. Furthermore, section IV details the proposed IEDA. Section V gives the experimental results and analysis of the two topologies. Finally, section VI provides conclusions.
respectively; $V_{SH}$ and $V_{LH}$ denote the RMS value of supply voltage and load voltage (line-to-neutral) at harmonic ‘$H$’, respectively. Fig. 1 (a) is a structure of an “APF in series with shunt passive filter”. The active filter eliminates the load harmonics by injecting harmonic current into the passive filter to improve the performance of the passive filter. Moreover, the basic system voltage across the passive filter will be reduced, thereby reducing the rated voltage of the APF [54]. Fig. 1 (b) is a diagram of “combined series APF and shunt passive filter” in which a series APF uses high impedance to provide harmonics and forces harmonic currents to flow to the passive filter, thereby reducing the current rating of the APF. In a power system, a point of common coupling (PCC) is generally considered to be the closest point to the user, and the system owner or operator can provide services to another user [55]. Here, it is identified as the point at which other linear loads are connected to the system. A passive filter is a set of tuned filters. $X_L$ and $X_C$ are a set of tuned filters or simple single tuned filters, depending on the requirements of the system.

Fig. 2 shows a single-phase equivalent circuit for a HAPF configuration of the fundamental frequency. The subscript “1” of the parameter in the figure indicates the value of the parameter at the fundamental frequency. The single-phase equivalent circuits of the two configurations are different in harmonic frequency, as shown in Fig. 3 and Fig. 4, which respectively represent “APF in series with shunt passive filter” and “combined series APF and shunt passive filter”. The difference between these two configurations is because of the location of the active power filter and how it reacts to

---

**FIGURE 1.** Circuit configurations of HAPF.

![Circuit configurations of HAPF](image1.png)

(a) Config.1: APF in series with shunt passive filter.

(b) Config.2: Combined series APF and shunt passive filter.

**FIGURE 2.** Single-phase equivalent circuit at fundamental frequency ($H = 1$).

![Single-phase equivalent circuit at fundamental frequency](image2.png)

**FIGURE 3.** Single-phase equivalent circuit for config.1 'APF in series with shunt passive filter' at harmonic frequencies ($H \geq 2$).

![Single-phase equivalent circuit for config.1](image3.png)

**FIGURE 4.** Single-phase equivalent circuit for config.2 'combined series APF and shunt passive filter' at harmonic frequencies ($H \geq 2$).

![Single-phase equivalent circuit for config.2](image4.png)
power harmonics. The active power filter is considered to be a controlled voltage source (such as $V_{AF}$) whose performance is achieved by injecting a voltage harmonic waveform proportional to the supply current harmonic component (i.e. $V_{AF} = K_{sh}I_{sh}$) at its terminal. The filter gain $K$ is designed to provide zero impedance at the fundamental frequency. It implies that the active filter element acts as a virtual harmonic resistor. The focus of this study is on the optimization of $K$, $X_L$, $X_C$ in the case where both source and load are nonlinear. The source harmonic voltage and current non-linearity are calculated in $V_{SH}$ and $I_{SH}$, respectively, while the harmonic voltage and current nonlinearity of the load are considered in $V_{LH}$ and $I_{LH}$, respectively.

Thevenin voltage source representing the power supply voltage and the harmonic current source representing the nonlinear load are [18],

$$v_S(t) = \sum_{h} v_{sh}(t)$$

$$i_L(t) = \sum_{h} i_{lh}(t)$$

The $H$-th harmonic source impedance is

$$Z_{SH} = R_{SH} + jX_{SH}$$

The $H$-th harmonic load impedance is

$$Z_{LH} = R_{LH} + jX_{LH}$$

Accordingly, the load admittance is

$$Y_{LH} = G_{LH} - jB_{LH}$$

Analyze the equivalent circuit in Fig. 3 of config.1 “APF series with shunt passive filter”, and get the following equations for the compensated utility supply current and load voltage respectively for harmonic “$H \geq 2$”

$$I_{SH} = \frac{A + jB}{C + jD}$$

$$V_{LH} = \frac{E + jF}{C + jD}$$

Analyze equivalent circuit in Fig. 4 for config.2 “combined series APF and shunt passive filter”, and get the following equations for the compensated utility supply current and load voltage respectively for harmonic “$H \geq 2$”

$$I_{SH} = \frac{A + jB}{C + jD'}$$

$$V_{LH} = \frac{E + jF}{C + jD'}$$

where,

$$A = V_{SH}R_{LH} - I_{LH}X_{LH}(H_{LH} \cdot \frac{X_C}{H})$$

$$B = V_{SH} \left( X_{LH} + H_{LH} \cdot \frac{X_C}{H} \right) + I_{LH}R_{LH}(H_{LH} \cdot \frac{X_C}{H})$$

$$C = R_{TLH} + KR_{LH} - (X_{LH} + X_{SH})(H_{LH} \cdot \frac{X_C}{H})$$

$$R_{TLH} = R_{SH}R_{LH} - X_{SH}X_{LH}$$

$$D = X_{TLH} + KR_{LH} + (R_{SH} + R_{LH})(H_{LH} \cdot \frac{X_C}{H})$$

$$X_{TLH} = R_{LH}X_{SH} + R_{SH}X_{LH}$$

$$E = V_{SH} \left[ KR_{LH} - X_{LH} \left( H_{LH} \cdot \frac{X_C}{H} \right) \right] + I_{LH}X_{LH}(H_{LH} \cdot \frac{X_C}{H})$$

$$F = V_{SH} \left[ KX_{LH} - R_{LH} \left( H_{LH} \cdot \frac{X_C}{H} \right) \right] + I_{LH}R_{LH}(H_{LH} \cdot \frac{X_C}{H})$$

$$D' = X_{TLH} + KR_{LH} + (K + R_{SH} + R_{LH})(H_{LH} \cdot \frac{X_C}{H})$$

$$F' = V_{SH} \left[ KX_{LH} - (K + R_{LH}) \left( H_{LH} \cdot \frac{X_C}{H} \right) \right] + I_{LH}R_{LH}(H_{LH} \cdot \frac{X_C}{H})$$

Through a detailed review of formulas (6) and (8), it is found that the compensated power supply harmonic current, $I_{SH}$ and the gain $K$ are inversely proportional. The active filter acts as a “obstructing resistor” that blocks the harmonic current generated by the source nonlinear $V_{SH}$. In addition, it acts as a “damping resistor” for the harmonic current $I_{LH}$, which completely attenuates the resonance between the parallel passive filter and the source impedance.

On the other side, the compensated $V_{LH}$ has an extra proportional $K$ term in the numerator and an inverse term in the denominator, which can be seen in formulas (7) and (9). Therefore, the goal of optimization is to find a suitable gain $K$ value such that its value can simultaneously reduce $I_{SH}$ (i.e. $ITHD$) and $V_{LH}$ (i.e. $VTHD$), rather than being high enough to adversely affect $V_{LH}$. The calculation formulas of other system parameters are as follows:

**Compensated load displacement power factor (DPF),**

$$DPF = \frac{P_{L1}}{V_{L1}I_{L1}} = \frac{G_{L1}V_{L1}}{I_{S1}}$$

where, subscript “1” denotes the fundamental component.

**Compensated load power factor (PF),**

$$PF = \frac{P_L}{V_LI_S} = \frac{G_{L1}V_{L1} + \sum_{H=2} G_{LH}V_{LH}^2}{(I_{S1}^2 + \sum_{H=2} I_{SH}^2)(V_{L1}^2 + \sum_{H=2} V_{LH}^2)}$$

**Transmission loss is given by,**

$$P_{LOSS} = I_{S1}^2R_{S1} + \sum_{H=2} I_{SH}^2R_{SH}$$

**Transmission efficiency is calculated as,**

$$\eta = \frac{P_L}{P_L + P_{LOSS}}$$

**Compensated $VTHD$ at the load terminal,**

$$VTHD = \sqrt{\sum_{H=2} V_{LH}^2}$$
Compensated $ITHD$ for the utility supply current,

$$ITHD = \sqrt{\frac{\sum_{n \geq 2} I_n^2}{V_{s1}}}$$ (25)

Approximated formula for Harmonic Pollution ($HP$) as per [56],

$$HP = \sqrt{VTHD^2 + ITHD^2}$$ (26)

B. OBJECTIVE FUNCTION

In HAPF, the optimized filter parameters are $K$, $X_L$, $X_C$, and each variable is in the range of ohms.

- $0 \leq K \leq 20$
- $0 \leq X_C \leq 10$
- $0 \leq X_L \leq 1$

The final objective function of this paper is to minimize $HP$. In accordance with the IEEE standard, we introduce the objective function as follows:

To ensure that $VTHD$ and $ITHD$ are tolerable, i.e. $VTHD \leq VTHD_{lim}$ and $ITHD \leq ITHD_{lim}$, the objective function of the optimization is formulated as:

$$HP_{APP} = \text{abs}(VTHD_{lim} - VTHD) + \text{abs}(ITHD_{lim} - ITHD)$$ (27)

where,

- $VTHD_{lim}$ = limitation on $VTHD$ prescribed by IEEE 519-2014 [55] based on system voltage level.
- $ITHD_{lim}$ = limitation on $ITHD$ prescribed by IEEE 519-2014 [55] based on system short circuit ratio.

The purpose of optimization while satisfying a single harmonic within the scope of the IEEE standard:

Maximize ‘$HP_{APP}$ subject to $PF_{PF} = PF_{goal} \pm \varepsilon$’ (28)

where, the power factor and $\varepsilon$ required for $PF_{goal}$ are a small error value ($< 10^{-3}$) to facilitate the iterative process. Therefore, the maximization of the objective function will ensure that both $VTHD$ and $ITHD$ are minimized, and these values will be far from the defined limits of $VTHD_{lim}$ and $ITHD_{lim}$, and approach $0$. Therefore, “$-HP_{APP}$” (negative) is used as the objective function input in the algorithm, and it is minimized by optimizing $K$, $X_L$, and $X_C$. In addition, the algorithm can only find the value of the objective function if all constraints of individual and total harmonic levels are met. Otherwise, the objective function returns a random high value $OBJ_{temp}$, discarding the set of decision variables that caused the constraint violation in subsequent iterations. In this case, choose a random positive value of $OBJ_{temp} = 1$, because a positive value of “$-HP_{APP}$” is not feasible when evaluating any set of values for $K$, $X_L$, and $X_C$. In terms of circuit resonance, this phenomenon means higher harmonic pollution and losses.

By minimizing $HP_{APP}$, we can get the optimal solution ($K^*, X^*_L \& X^*_C$) that meets the IEEE standard. We import the obtained optimal solution into $HP$ and get the optimal $HP$ value. $HP$ can be defined as:

$$HP = HP(K^*, X^*_L \& X^*_C)$$ (29)

where, ($K^*, X^*_L \& X^*_C$) is the parameter value that minimizes $HP_{APP}$. If the solution that meets the IEEE standard is not obtained during the optimization process, the final objective function $HP$ will get a meaningless random value. In this case, we set $HP$ to an impossible value ($HP = 10\%$).

C. CASE STUDIES

TABLE 1 lists the conditions for the various case studies performed in this paper for these two HAPF configurations. The first 3 cases are mentioned in [33]. The case 4 is mentioned by [39], and case 4 has a higher source of harmonic pollution. The numerical data used is a power plant with a total apparent load of three phases ($5100 + j4965$) kVA at a line voltage of $4.16kV$. The short-circuit capacity of the system is $80MVA$. The source and load harmonics are assumed to be time-invariant quantities; load and source resistances are independent of frequency i.e. $R_{SH} = R_L$ and $R_{SH} = R_S$. $PF_{goal}$ is selected as $95\%$ if not mentioned otherwise. Both $VTHD_{lim}$ and $ITHD_{lim}$ are $5\%$ based on the system under study.

| Parameters | Case 1 | Case 2 | Case 3 | Case 4 |
|------------|--------|--------|--------|--------|
| $R_{S1}$ (Ω) | 0.02163 | 0.02163 | 0.02163 | 0.02163 |
| $X_{S1}$ (Ω) | 1.7421 | 1.7421 | 1.7421 | 1.7421 |
| $X_{L1}$ (Ω) | 1.696 | 1.696 | 1.696 | 1.696 |
| $V_S$ (kV) | 2.40 | 2.49 | 2.40 | 2.40 |
| $V_{S8}$ (%$V_{S1}$) | 0.0 | 0.0 | 0.0 | 0.0 |
| $V_{S7}$ (%$V_{S1}$) | 0.0 | 0.0 | 0.0 | 0.0 |
| $V_{S11}$ (%$V_{S1}$) | 0.0 | 0.0 | 0.0 | 0.0 |
| $I_{L8}$ (%$I_L$) | 40.0 | 40.0 | 40.0 | 40.0 |
| $I_{L7}$ (%$I_L$) | 6.0 | 6.0 | 6.0 | 6.0 |
| $I_{L13}$ (%$I_L$) | 1.0 | 1.0 | 1.0 | 1.0 |

III. THE ORIGINAL DRAGONFLY ALGORITHM

DA is a new meta-heuristic based algorithm based on dragonfly behavior. It consists of five parts: separation, alignment, cohesion, enemy distraction, and food attraction.

Each of these behaviors is mathematically modeled as follows.

Separation:

$$S_i = -\sum_{j=1}^{N} X - X_j$$ (30)

where, $X$ is the current position of the dragonfly, $X_j$ is the position of the $j$th dragonfly adjacent to the $X$ dragonfly,
and \(N\) is the total number of individuals adjacent to the \(X\) dragonfly.

Alignment:

\[
A_i = \frac{\sum_{j=1}^{N} V_j}{N} \tag{31}
\]

where, \(V_j\) refers to the speed of the \(j\)-th neighboring individual.

Cohesion:

\[
C_i = \frac{\sum_{j=1}^{N} X_j}{N} - X \tag{32}
\]

Enemy distraction:

\[
E_i = X + X^- \tag{33}
\]

where, \(X^-\) refers to the position of the natural enemy.

Food attraction:

\[
F_i = X^+ - X \tag{34}
\]

where, \(X^+\) refers to the location of the food.

The food source position is the current optimal position of the algorithm, and the natural enemy position is the current worst position. The dragonfly’s behavior is formed by the combination of the above five correction methods. The author proposes a step vector and a position vector to imitate the position of the dragonfly by imitating the particle swarm optimization (PSO).

The step vector update is obtained by the formula:

\[
\Delta X^{t+1} = w\Delta X^t + (sS_i + aA_i + cC_i + fF_i + eE_i) \tag{35}
\]

where, \(S_i, A_i, C_i, F_i, E_i\) refer to the above five correction methods, \(\omega\) is the inertia weight, and \(s, a, c, f,\) and \(e\) respectively refer to separation weight, formation weight, aggregation weight, predation weight, natural enemy weight, \(t\) represents the current number of iterations.

It is worth noting that if there are no neighbors around the particle, the current dragonfly will use Lévy-Flight strategy in the search space. In this case, the update method is as follows:

When \(N > 0\),

\[
X^{t+1} = X^t + \Delta X^{t+1} \tag{36}
\]

When \(N = 0\), Lévy-Flight is used in the search space.

\[
X^{t+1} = \text{Levy}(d) \times X^t + X^t \tag{37}
\]

where, \(d\) is the dimension of the dragonfly and \(t\) represents the current number of iterations.

Lévy-Flight is calculated as follows:

\[
\text{Levy}(x) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^\beta} \tag{38}
\]

\[
\sigma = \left( \frac{\Gamma (1 + \beta) \times \sin \left( \frac{\pi \beta}{2} \right)}{\Gamma (\frac{1 + \beta}{2})} \times \beta \times 2^{\left( \frac{\beta - 1}{2} \right)} \right) \tag{39}
\]

where, \(\beta\) is a constant and taken to be 1.5 in this paper, \(r_1\) and \(r_2\) is a random number between 0 and 1, and \(\Gamma (x) = (x - 1)!\).

IV. THE ALGORITHM OF THE IEDA

A. EXEMPLAR POOL STRATEGY

The original DA has excellent global search capability, but because the information of the superior particles of the previous generation does not communicate with the next generation of particles, the superior particles cannot guide particle updates, resulting in weak algorithm exploitation capability. Consequently, we added the exemplar pool strategy to the IEDA and used the exemplar pool (EP) to store the high-quality particle information in the algorithm iteration process and guide the particles to update. An exemplar pool can be represented by a collection as follows:

\[
EP = \{E_1, E_2, E_3, \ldots, E_{NP/2}\} \tag{40}
\]

where, \(NP\) is the total number of particles. We sort personal best value of all particles and load the top half of the particle information into the exemplar pool. \(E_i\) can be expressed as:

\[
E = \text{sort}(F(pb_1), F(pb_2), F(pb_3), \ldots, F(pb_{NP})) \tag{41}
\]

where, \(pb_i\) is the personal best position of the \(i\)-th particle, \(F(\cdot)\) denotes the fitness function, and \(\text{sort}()\) is to sort \(pb_{best_i}\) from the best to the worst.

B. DIVISION OF LABOR STRATEGY

In the IEDA, we use the clustering method to divide particles into exploitation population and exploration population. The position of the dragonfly in the search space of the DA is randomly distributed. When there are no nearby particles around the current particle, the particle executes the Lévy-Flight strategy, which makes the DA have excellent global search capability, so we will use DA to update the exploration population. In the exploitation population, exploitation population can take advantage of the high-quality particle information, which provided by the exploration population to the example pool. Then, the exploitation can quickly converge to the optimal value.

1) EXPLOITATION POPULATION

In the exploitation population, we borrowed the information exchange mechanism of the DE and introduced the update formula of “DE/current-to-best/1”. The random particles in the update formula can be selected from the exemplar pool or the exploitation population. Because the quality of particles in exemplar pool perform poorly in the early stage of algorithm iteration, the exemplar pool cannot provide high-quality particle information to the exploitation population. We have introduced an exploitation ability probability curve (\(Pc\)), as shown in Fig. 5. The \(Pc\) curve can control the exploitation ability of the exploitation population by controlling the selection of random particles during the entire iteration. So that the exploitation population has a certain exploration ability in the early iteration. The relationship between the exploitation ability probability curve \(Pc\) and the number of evaluations can be expressed as:

\[
Pc = a + b \times \frac{(\exp (3 (nfes - 1)) / (\max nfes - 1) - 1)}{(\exp (3) - 1)} \tag{42}
\]
The updated formula for the exploration population can be expressed as:

\[
X_i^{G+1} = X_i^G + F \cdot (X_{gbest}^G - X_i^G) + F \cdot \left( X_{r_3}^G - X_{r_4}^G \right)
\]

if \( \text{rand()} \geq P_c(nfes) \)

\[
X_i^{G+1} = X_i^G + F \cdot (X_{gbest}^G - X_i^G) + F \cdot (EP_{r_3}^G - EP_{r_4}^G)
\]

if \( \text{rand()} < P_c(nfes) \)

(43)

where, ‘nfes’ represents the number of fitness evaluations, ‘maxnfes’ represents the maximum number of fitness evaluations, and the constants of \( a \) and \( b \) are 0.05 and 0.45, respectively [53].

The updated formula for the exploitation population can be defined as:

\[
\begin{align*}
X_i^{G+1} &= X_i^G + F \cdot (X_{gbest}^G - X_i^G) + F \cdot (X_{r_3}^G - X_{r_4}^G) \\
&\text{if } \text{rand()} \geq P_c(nfes)
\end{align*}
\]

\[
\begin{align*}
X_i^{G+1} &= X_i^G + F \cdot (X_{gbest}^G - X_i^G) + F \cdot (EP_{r_3}^G - EP_{r_4}^G) \\
&\text{if } \text{rand()} < P_c(nfes)
\end{align*}
\]

(44)

where, the scale factor \( F \) is a positive control parameter for scaling the difference vector and \( F \) is a constant and taken to be 0.9, which is a good compromise between speed and probability of convergence [57]. \( r_3 \) and \( r_4 \) are randomly selected integers from 1 to \( NP/2 \). And, \( \text{rand()} \) is a random number between 0 and 1. \( P_c(nfes) \) is the \( P_c \) value when the number of fitness evaluations is ‘nfes’.

2) EXPLORATION POPULATION

We use the global search ability of the DA in the exploration population, and the DA stores the obtained superior particle information in the exemplar pool. The exploitation population completes the information exchange between the two populations by using the information in the exemplar pool, so that the algorithm avoids falling into the local optimum. Because the exploitation capability of the exploitation population is strong enough, we remove the exploitation ability of the DA to improve the exploration ability and operation speed. The updated formula for the exploitation population can be expressed as:

\[
\Delta X_{i}^{t+1} = w \Delta X_{i}^{t} + (sS_{i} + aA_{i} + cC_{i})
\]

(44)

\( S_{i}, A_{i}, C_{i} \) refer to the above three correction methods, \( w \) is the inertia weight, \( s, a \) and \( c \) respectively refer to separation weight, formation weight, aggregation weight, \( t \) represents the current number of iterations.

Then, the fuzzy distance judgment mechanism is used to change the distance judgment of the DA from comparing distances to comparing distances in a single dimension, which simplifies the calculation cost of the distance and improves the calculation speed.

C. TIME COMPLEXITY ANALYSIS OF IEDA

The time complexity of the original DA algorithm mainly depends on two parts, namely compute the number of neighboring individual and update the position of current particle. Thus, the time complexity can be defined as follows:

\[
O(\text{IEDA}) = O(\text{compute neighbors}) + O(\text{position update})
\]

(45)

\[
O(\text{IEDA}) = O(t \left( n^2 \times d + n \times d \right)) = O(tn^2)
\]

(46)

where, \( t \) denotes the maximum number of iterations, \( n \) presents the number of particles, and \( d \) is the dimension of given problem.

The proposed algorithm is divided into two populations. The time complexity of exploration population mainly depends on two processes: compute the number of neighboring individual and update the position of current particle. The time complexity of exploitation population depends only on the process of updating the position of the current particle. Due to the use of fuzzy distance judgments in exploration populations to calculate the number of neighboring particles through one dimension, i.e., the value of \( d \) is 1. Thus, the total time complexity can be defined as follows:

\[
O(\text{IEDA}) = O(\text{exploitation population}) + O(\text{compute neighbors}) + O(\text{position update})
\]

(47)

\[
O(\text{IEDA}) = O(t \left( n^2 \times d + n \times d \right)) + O(n^2) = O(tn^2)
\]

(48)

\[
O(\text{IEDA}) = O(t \left( n^2 \times d + n \times d \right)) = O(tn^2)
\]

(49)

where, \( t \) denotes the maximum number of iterations and \( n \) presents the number of particles, and \( d \) is the dimension of given problem.

It can be seen that the time complexity of the IEDA algorithm and the DA algorithm mainly depends on the calculation process of the number of neighboring individuals. But because IEDA only uses one dimension to calculate the number of neighboring individuals. Therefore, under the same conditions, the time complexity of the IEDA algorithm is \( d \) times lower than that of DA.

D. PROCEDURE OF IEDA

The process of IEDA can be described by pseudo code and flowchart (as shown in Fig. 6 below).

V. THE RESULTS AND ANALYSIS OF EXPERIMENT

To evaluate the performance of IEDA algorithm and other meta-heuristic algorithms on this problem, we have selected Dragonfly algorithm (DA) [40], particle swarm
optimization (PSO) [15], improved DA (IDA) [27], differential evolution (DE) [16], Moth-Flame Optimization (MFO) [14] a double-evolutionary learning MFO (DELMFO) [20], an approach based on differential evolution (DE) (L-SHADE) [39] and IEDA to compare. It is worth mentioning that L-SHADE is an algorithm for HAPF parameter
Algorithm 1 “IEDA Algorithm”

1: /* Initialization */
2: Randomly initialize each individual $X_i, i \in \{1, 2, 3, \cdots, Np\}$
3: Calculate constraints $VTHD, ITHD, PF$ for $X_i$, where $i=1$ to $Np$
4: if constraints are satisfied
5: Evaluate $f(X_i)$ i.e. “$-HP_{APP}$” for $X_i$, where $i=1$ to $Np$. Increase counter $nfes$ by $Np$ i.e. $nfes= nfes + Np$
6: else
7: Return $f(X_i) = OBJ_{temp}$. Increase counter $nfes$ by $Np$ i.e. $nfes= nfes + Np$
8: end if
9: Initialization personal best value ($pbest$), global best value ($gbest$)
10: Select exemplar particles for exemplar pool using formula (40) and formula (41)
11: while ($nfes <= maxnfes$) do
12: /* exploitation population */
13: for $i=1:NP/2$
14: Update the position of current particle using formula (43)
15: end for
16: /* Exploration population */
17: for $i=(NP/2+1):NP$
18: Calculate the number of neighboring individuals($N$), i.e. the number of particles which are within a radius of the current particle
19: if $N=0$
20: Update the position of current particle using formula (37)
21: else
22: Update the position of current particle using formula (44)
23: end if
24: end for
25: Correct the current particle position if it is out of bounds.
26: Calculate constraints $VTHD, ITHD, PF$ for $X_i$, where $i=1$ to $Np$
27: if constraints are satisfied
28: Evaluate $f(X_i)$ i.e. “$-HP_{APP}$” for $X_i$, where $i=1$ to $Np$. Increase counter $nfes$ by $Np$ i.e. $nfes= nfes + Np$
29: else
30: Return $f(X_i) = OBJ_{temp}$. Increase counter $nfes$ by $Np$ i.e. $nfes= nfes + Np$
31: end if
32: Update personal best value ($pbest$), global best value ($gbest$)
33: Select exemplar particles for exemplar pool using formula (40) and formula (41)
34: end while

optimization proposed in [39]. For the HAPF compensation configuration system, these algorithms are applied to the four study cases listed in TABLE. 1. The user defined control parameters of each algorithm are shown in TABLE. 2. These parameters were selected after several algorithm experiments. For the above methods, the population size and the maximum evaluation number are set to 50 and 50,000, respectively. To ensure the reliability of our statistical results, we run each study case 30 times and get the statistical results. In addition, $HP$ in formula. (29) was used to evaluate the accuracy and robustness of each algorithm. For all of the algorithms, the overall best $HP$ results are highlighted in bold.

**TABLE 2.** The control parameters of various optimization algorithms.

| Algorithm | Control parameters | Value |
|-----------|--------------------|-------|
| IEDA      | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Inertia weight ($w$) | 0.9-0.4 |
|           | Scale factor ($F$) | 0.9    |
| PSO       | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Inertia weight ($w$) | 0.9-0.4 |
|           | Cognitive learning factor ($c1$) | 2     |
|           | Social learning factor ($c2$) | 2     |
| DA        | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Inertia weight ($w$) | 0.9-0.4 |
| IDA       | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Inertia weight ($w$) | 0.9-0.4 |
|           | Mutation scaling factor ($SF$) | 0.5   |
|           | Crossover probability ($CR$) | 0.9   |
| DE        | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Scale factor ($F$) | 0.5    |
|           | Crossover rate ($CR$) | 0.9    |
| MFO       | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
| DELMFO    | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |
|           | Scale constant ($\sigma$) | 0.15  |
|           | Constant factor ($\theta$) | 0.6   |
|           | Crossover rate ($CR$) | 0.7   |
| LSHADE    | Population size ($Np$) | 50    |
|           | Maximum number of fitness evaluations | 50000 |

A. HAPF CONFIG. 1
Harmonic pollution ($HP$) is the objective function of our optimization problem. TABLE 3 lists the experimental results
of eight algorithms, including the minimum HP value (Best), maximum HP value (Worst), average HP value (Mean), and deviation of HP values (Std. dev.), and pass rate. mean, and std. dev. represent the accuracy, average accuracy, and reliability of the evaluation parameters of the corresponding comparison method, respectively. The pass rate percentage is calculated based on the number of times that an algorithm successfully obtains a feasible parameter in 30 runs. The feasible parameter refers to the parameter obtained when 
\[ -HP_{APP} \] is not equal to 1, not the optimal value. If the difference between the two values of the result is less than \(10^{-3}\), the two objective values are considered to be same.

It can be seen from TABLE 3 that in terms of average accuracy and reliability, the IEDA proposed in this paper is significantly better than most comparison algorithms of the four case types. For case 1, IEDA has the best results in terms of accuracy, average accuracy, and reliability. In case 2, the accuracy of the DA is the highest, but its average accuracy is the worst compared to other algorithms. The accuracy of the IEDA is slightly worse, but its average accuracy is the best.

In case 3, IEDA obtain the best results in terms of accuracy, average precision, and reliability. Interestingly, the IEDA can reach the optimal value at one time. Especially for case 4, case 4 has a higher harmonic pollution source, many algorithms cannot get a feasible solution. But the IEDA also successfully obtained the best results in terms of accuracy and average accuracy. And DE is also a powerful competitive algorithm, which is only slightly worse in reliability than IEDA.

In addition, Fig. 7 shows a box plot of each comparison methods in 30 independent runs. The symbol ‘+’ in the figure represents an outlier. From the box plots, we can see that IEDA excels in accuracy and reliability.

Fig. 8 shows the 
\[ -HP_{APP} \] performance convergence curve of the algorithm in 30 independent runs. IEDA is an algorithm that enhances the population exploration ability by slowing down the convergence speed, so the convergence speed of IEDA is not fast in the early iterative process. However, for these four cases, IEDA rarely falls into the local optimum.

| Case no. | Algorithm | Harmonic Pollution (in %) | Wilcoxon signed-rank test |
|----------|-----------|---------------------------|--------------------------|
|          |           | Best | Worst | Mean | Std. dev. | Pass rate (%) |
| Case 1   | IEDA      | 0.236 | 0.493 | 0.244 | 0.0469 | 100 |
|          | PSO       | 0.241 | 0.503 | 0.471 | 0.0761 | 100 + |
|          | DA        | 0.236 | 0.499 | 0.443 | 0.1028 | 100 + |
|          | IDA       | 0.236 | 0.493 | 0.322 | 0.1233 | 100 + |
|          | DE        | 0.236 | 0.493 | 0.347 | 0.1296 | 100 + |
|          | MFO       | 0.236 | 0.493 | 0.416 | 0.120 | 100 + |
|          | DELMFO    | 0.236 | 0.493 | 0.270 | 0.0880 | 100 ≈ |
|          | L-SHADE   | 0.236 | 0.493 | 0.304 | 0.1157 | 100 ≈ |
| Case 2   | IEDA      | 2.752 | 2.752 | 2.752 | 5.821e-05 | 100 |
|          | PSO       | 2.748 | 2.972 | 2.928 | 0.0681 | 100 + |
|          | DA        | 2.721 | 3.001 | 2.892 | 0.0953 | 100 + |
|          | IDA       | 2.752 | 2.952 | 2.812 | 0.0934 | 100 + |
|          | DE        | 2.752 | 2.952 | 2.839 | 0.1010 | 100 + |
|          | MFO       | 2.752 | 2.952 | 2.812 | 0.0934 | 100 + |
|          | DELMFO    | 2.752 | 2.952 | 2.785 | 0.0760 | 100 ≈ |
|          | L-SHADE   | 2.752 | 2.952 | 2.785 | 0.0760 | 100 ≈ |
| Case 3   | IEDA      | 5.672 | 5.672 | 5.672 | 4.063e-11 | 100 |
|          | PSO       | 5.672 | 5.893 | 5.860 | 0.0743 | 100 + |
|          | DA        | 5.675 | 5.899 | 5.862 | 0.0709 | 100 + |
|          | IDA       | 5.672 | 5.893 | 5.724 | 0.0933 | 100 + |
|          | DE        | 5.672 | 5.888 | 5.809 | 0.1059 | 100 + |
|          | MFO       | 5.672 | 5.888 | 5.856 | 0.076 | 100 + |
|          | DELMFO    | 5.672 | 5.888 | 5.816 | 0.1036 | 100 + |
|          | L-SHADE   | 5.672 | 5.888 | 5.837 | 0.0878 | 100 + |
| Case 4   | IEDA      | 6.340 | 6.340 | 6.340 | 3.193e-15 | 100 |
|          | PSO       | 6.341 | 6.981 | 6.602 | 0.3448 | 16.67 + |
|          | DA        | 6.340 | 6.981 | 6.625 | 0.3255 | 60.00 + |
|          | IDA       | 6.340 | 6.979 | 6.514 | 0.2979 | 36.67 + |
|          | DE        | 6.340 | 6.340 | 6.340 | 3.2066e-15 | 100 ≈ |
|          | MFO       | 6.340 | 7.041 | 6.646 | 0.317 | 53.33 + |
|          | DELMFO    | 6.340 | 6.858 | 6.438 | 0.1612 | 80.00 + |
|          | L-SHADE   | 6.340 | 7.012 | 6.453 | 0.2150 | 80.00 + |
In addition, we adopted the Wilcoxon signed rank test with a significance level of 0.05 to compare IEDA with other comparisons [59]. Labels “+”, “−”, and “≈” indicate that IEDA is significantly better than, worse than, and similar to the compared with methods. From the results of the Wilcoxon signed rank test shown in Table 3, it can be seen that in the configuration 1 of the hybrid power active filter, IEDA is superior to most comparison algorithms. Although in some cases, the data obtained by the comparison algorithm and the IEDA algorithm are similar, but overall, the IEDA algorithm is superior to other comparison algorithms.

The circuit configuration optimization results and calculated values are shown in Table 4 for config.1. The optimized passive filter inductive reactance, capacitive reactance value and filter feedback gain $K$ value are basically consistent with previous research results. The results show that when the system load and the basic impedance of the source are unchanged, the higher the gain, the better the compensation effect, until the nonlinearity of the source becomes significant. In general, the higher the control gain of an active filter (which is inversely proportional to the resistance of the harmonic frequency), the lower the voltage harmonics appearing on its installation bus [37]. Although the larger gain was found to have a smaller effect on compensating $VTHD$ than it indicated in [18]. Case 4 emphasizes the sensitivity of the optimal value of $K$ to power supply distortion. In Case 4, the 11th and 13th voltage harmonics of the source are made slightly higher; however, this results in a significantly lower optimal value of $K$. A possible explanation can be derived from equations (7) and (9). The feedback gain $K$ is closely related to $V_{SH}$ and the numerator becomes dominant. Therefore, the gain $K$ must be selected correctly, especially in the case of high voltage distortion.
Further review of TABLE. 4 reveals that as the level of harmonics in the power voltage increases, harmonic pollution is on the rise. This is due to the increase in both $VTHD$ and $ITHD$. Besides, higher utility distortion forces the gain $K$ to be lower and $X_L$ to almost zero. It represents a low rated voltage source active power filter in series with a passive filter and does not require additional switching filters to eliminate current fluctuations.

Fig. 9 (a) and Fig. 9 (b) respectively show the harmonic content of the load voltage $(V_{\text{LH}})$ and supply current $(I_{\text{SH}})$ compensated by HAPF config. 1 “APF in series with shunt passive filter” under different conditions in bar graphs. Obviously, all harmonics are within the allowable range specified by the standard IEEE 519 [55].

B. HAPF CONFIG. 2

As with config. 1, TABLE. 5 shows the experimental results of eight algorithms in config. 2, including the best, worst, average, standard deviation, and pass rate when each algorithm was run 30 times. And because the fundamental frequency equivalent circuits of the two HAPF structures are the same and the two configurations use the same parameters, the result of HAPF config. 2 is similar to config. 1. According to TABLE. 5, IEDA is superior to most algorithms in accuracy, average accuracy, and reliability. For case 1 and case 3, terms of the accuracy, average accuracy, and reliability of IEDA are all optimal, and they all reach the optimal value at one time. For case 2, DA obtained better results in accuracy, but IEDA was optimal in average accuracy and reliability. In case 4, DE is optimal in terms of accuracy, average accuracy, and reliability. But the most powerful competitor, IEDA also achieved satisfactory results in addition to reliability. And many competitive algorithms cannot obtain feasible solutions that meet IEEE standards every time, and the results obtained also perform poorly in accuracy, average accuracy, and reliability. As with config. 1, we give box plots of each comparison method in 30 independent runs in Fig. 10.

Similar optimization studies were performed on HAPF config. 2 of “combined series APF and shunt passive filter”. The results of all study cases are shown in TABLE. 6. As can be seen from the data in the table, the calculation results are basically consistent with the circuit configuration 1. A careful study of the equations related to circuit configuration 2 will help to find the cause of the close match of the output results. The fundamental frequency equivalent circuits of the two HAPF structures are the same. Due to the existence of active filters, there are differences in equivalent circuits at harmonic frequencies. The nuances of the denominator formulas (6) and (8) compensate the utility supply current $(I_{\text{SH}})$ for the two HAPF configurations. However, the load inductive reactance at the harmonic frequency is dominant, and small changes in other parameters have no significant effect on the power supply current harmonics. Comparing the two HAPF configurations of formulas (7) and (9), for the compensated load voltage $(V_{\text{LH}})$, equation (9) adds a term to both the numerator and denominator, and $K$ times the harmonic impedance of the passive filter. Like as in $I_{\text{SH}}$, this is much smaller than the reactive load. The additional terms are changed at the harmonic frequency and the numerator and denominator, and the complex $V_{\text{LH}}$ has almost no effect. Therefore, in the medium voltage system studied, two HAPF configurations can have almost the same best parameters.
TABLE 5. Comparisons of the results using different algorithms for case studies with HAPF config. 2.

| Case no. | Algorithm | Harmonic Pollution (in %) | Wilcoxon signed-rank test |
|----------|-----------|---------------------------|---------------------------|
|          |           | Best | Worst | Mean | Std. dev. | Pass rate (%) |
| Case 1   | IEDA      | 0.227 | 0.227 | 0.227 | 1.575e-06 | 100 +         |
|          | PSO       | 0.231 | 0.465 | 0.424 | 0.0844    | 100 +         |
|          | DA        | 0.227 | 0.461 | 0.358 | 0.1121    | 100 +         |
|          | IDA       | 0.227 | 0.459 | 0.328 | 0.117    | 100 +         |
|          | DE        | 0.227 | 0.459 | 0.351 | 0.1174    | 100 +         |
|          | MFO       | 0.227 | 0.459 | 0.389 | 0.108    | 100 +         |
|          | DELMFO    | 0.227 | 0.459 | 0.258 | 0.0800    | 100 ≈         |
|          | L-SHADE   | 0.227 | 0.459 | 0.320 | 0.1153    | 100 +         |
| Case 2   | IEDA      | 2.754 | 2.754 | 2.754 | 5.290e-05 | 100 +         |
|          | PSO       | 2.748 | 2.968 | 2.937 | 0.0485    | 100 +         |
|          | DA        | 2.704 | 2.958 | 2.919 | 0.0720    | 100 +         |
|          | IDA       | 2.752 | 2.949 | 2.852 | 0.0992    | 100 +         |
|          | DE        | 2.754 | 2.949 | 2.858 | 0.0989    | 100 +         |
|          | MFO       | 2.745 | 2.952 | 2.897 | 0.088    | 100 +         |
|          | DELMFO    | 2.754 | 2.949 | 2.793 | 0.0793    | 100 +         |
|          | L-SHADE   | 2.754 | 2.949 | 2.774 | 0.0595    | 100 +         |
| Case 3   | IEDA      | 5.681 | 5.681 | 5.681 | 3.023e-15 | 100 +         |
|          | PSO       | 5.681 | 6.034 | 5.898 | 0.0947    | 96.67 +       |
|          | DA        | 5.681 | 5.909 | 5.877 | 0.0765    | 100 +         |
|          | IDA       | 5.681 | 5.911 | 5.749 | 0.106    | 100 +         |
|          | DE        | 5.681 | 5.906 | 5.823 | 0.1104    | 100 +         |
|          | MFO       | 5.681 | 5.908 | 5.847 | 0.100    | 100 +         |
|          | DELMFO    | 5.681 | 5.906 | 5.846 | 0.1013    | 100 +         |
|          | L-SHADE   | 5.681 | 5.925 | 5.876 | 0.0780    | 100 +         |
| Case 4   | IEDA      | 6.370 | 6.370 | 6.370 | 1.067e-08 | 100 +         |
|          | PSO       | 6.373 | 7.060 | 6.512 | 0.3063    | 16.67 +       |
|          | DA        | 6.370 | 6.405 | 6.374 | 0.0099    | 40.00 +       |
|          | IDA       | 6.370 | 7.057 | 6.508 | 0.3070    | 16.67 +       |
|          | DE        | 6.370 | 6.370 | 6.370 | 1.8067e-15 | 100 ≈     |
|          | MFO       | 6.370 | 7.057 | 6.462 | 0.1883    | 50 +          |
|          | DELMFO    | 6.370 | 6.890 | 6.512 | 0.1735    | 76.67 +       |
|          | L-SHADE   | 6.370 | 6.470 | 6.377 | 0.0237    | 60.00 +       |

Fig. 12 (a) and Fig. 12 (b) respectively show the harmonic content of the load voltage ($V_{LH}$) and supply current ($I_{SH}$) compensated by HAPF config. 2 “combined series APF and shunt passive filter” under different conditions in bar graphs. All harmonics are within the range allowed by the IEEE 519 [55] standard.

C. EFFECTIVENESS OF THE INTRODUCED STRATEGIES

In this section, we take the four cases in configuration 1 as an example and do some experiments to verify the effectiveness of our proposed strategy. We have added the division of labor strategy, the exemplar pool strategy and the probability curve strategy in the IEDA algorithm, and we label them strategy 1, 2, and 3 in turn. These three strategies are introduced into the algorithm in sequence as needed. We give the convergence curves of the algorithms that sequentially introduce these three strategies. The convergence curve graphs use the average objective function value of 30 independent runs.

As shown in Fig. 13, the original DA does not perform well in these four cases. Through the division of labor strategy, the exploitation ability of the algorithm is improved to increase
the accuracy of the algorithm. However, there is still room for improvement. Then, we introduced the exemplar pool strategy to collect high-quality particle information and feed it back to the exploration population, so that the exploitation ability of the algorithm continues to be improved. Although this method makes the convergence speed faster, it is easy to fall into local optimum. Therefore, we introduced a probability curve to allow the exploitation population to retain a certain exploration ability. The probability curve slows the convergence rate, it improves the accuracy again. By adding these three strategies in turn, the exploitation ability of algorithm is enhanced under the condition that sufficient exploration ability is reserved. For case 4, the algorithm can get great results without adding strategy 3 and the convergence speed is faster, but considering these four cases, we still add strategy 3 to the algorithm.
In this section, some experiments are conducted to verify the rationality of the parameters \((F, a, b)\) in our algorithm, and discuss the effect of parameters on the performance of the algorithm. The scale factor \(F\) is a positive control parameter for scaling the difference vector, and the effective value of \(F\)
is usually selected in (0.4, 1). a and b are used to calculate the parameters of the Pc curve, and used to control the source of particles that communicate with the exploitation population. We adjust the parameters of F, a and b to verify the effect on the algorithm performance. As shown in Table. 7, when \( F = 0.9, a = 0.05, b = 0.45 \), the algorithm performance is the best. When F is equal to 0.9, the increase of a and b leads to the decrease of algorithm performance. In addition, we can intuitively draw a conclusion that F has a great influence on the algorithm, and the performance of the algorithm becomes worse as F decreases.

VI. CONCLUSIONS AND FUTURE WORK

This paper proposes a new meta-heuristic algorithm for parameter estimation of two topologies of hybrid power active filters. In the parameter estimation of the two topologies of hybrid power active filters, the performance of IEDA have been fully verified by comparison with other heuristic algorithms. Through comprehensive data comparison, the IEDA performs well, especially in terms of average accuracy. This technology helps to make a more accurate preliminary selection of the parameters of hybrid power active filters before field experimental verification.

However, due to the high time complexity of the IEDA algorithm, the IEDA algorithm is still worth further improvement. In the future, we will try to develop a periodic distance judgment method to reduce the number of distance calculations in the iterative process, thereby reducing the calculation cost of the algorithm and increasing the running speed. In addition, we will try to develop a better information feedback mechanism to further improve the performance of IEDA. At the same time, IEDA will be used to solve constraint multi-objective problems or more complex multi-dimensional problems in the future. At the same time, IEDA will be used to solve constraint multi-objective problems or more complex multi-dimensional problems in the future.

REFERENCES

[1] S. H. E. Abdel Aleem, A. F. Zobaab, and M. M. Abdel Aziz, “Optimal C-type passive filter based on minimization of the voltage harmonic distortion for nonlinear loads,” IEEE Trans. Ind. Electron., vol. 59, no. 1, pp. 281–289, Jan. 2012.
[2] W. Su, Z. Li, and Z. Shao, “The harmonic calculation model of non-linear load,” in Proc. China Int. Conf. Electr. Distribution (CICED), Shenzhen, China, Sep. 2014, pp. 866–872.
[3] L. Motta and N. Fatimé, “Active/passive harmonic filters: Applications, challenges & trends,” in Proc. 17th Int. Conf. Harmonics Qual. Power (ICHQP), Belo Horizonte, Brazil, 2016, pp. 657–662.
[4] Y. Xiu, X. Xiao, H. Liu, and H. Wang, “Parallel operation of hybrid active power filter with passive power filter or capacitors,” in Proc. IEEE/PES Transmiss. Distrib. Conf. Expo., Asia Pacific, Aug. 2005, pp. 1–6.
[5] Q. Chen, Z. Chen, and M. McCormick, “The application and optimization of C-type filter in a combined harmonic power filter,” in Proc. IEEE, 55th Annu. Power Electron. Spec. Conf., Aachen, Germany, vol. 2, Jun. 2004, pp. 1041–1045.
[6] E. Yang, Z. Wang, H. Wang, and G. Chen, “A new hybrid active power filter topology with continuous compensation bandwidth and small active capacity,” in Proc. 23rd Annu. IEEE Appl. Power Electron. Conf. Expo., Austin, TX, USA, Feb. 2008, pp. 1158–1162.
[7] B. Kedira, “Reducing inverter power rating in active power filters using proposed hybrid power filter topology,” in Proc. IEEE 15th Int. Conf. Environ. Electr. Eng. (EICEE), Rome, Italy, Jun. 2015, pp. 443–448.
[8] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, “GSA: A gravitational search algorithm,” Inf. Sci., vol. 179, no. 13, pp. 2232–2248, Jun. 2009.
[9] H. Fethia, T. Abdelhalim, A. Amel, M. Ahmed, and B. M. Sehghir, “Tuning gain of PI controller based on meta-heuristic techniques,” in Proc. Int. Conf. Appl. Smart Syst. (ICASS), Medea, Algeria, Nov. 2018, pp. 1–6.
[10] A. Jamil, N. Javaid, and S. Aslam, “An efficient home energy optimization by using meta-heuristic techniques while incorporating game-theoretic approach for real-time coordination among home appliances,” in Proc. 5th Int. Multi-Topic ICT Conf. (IMITIC), Jamshoro, Pakistan, Apr. 2018, pp. 1–6.
[11] G. G. Wang, S. Deb, and Z. Cui, “Monarch butterfly optimization,” Neural Comput. Appl., vol. 31, no. 7, pp. 1995–2014, 2019.
[12] G. G. Wang, “Moth search algorithm: A bio-inspired metaheuristic algorithm for global optimization problems,” Memetic Comput., vol. 10, no. 2, pp. 151–164, Jun. 2018.
[13] S. Mirjalili, “SCA: A sine cosine algorithm for solving optimization problems,” Knowl.-Based Syst., vol. 96, pp. 120–133, Mar. 2016.
[14] S. Mirjalili, “Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm,” Knowl.-Based Syst., vol. 89, pp. 228–249, Nov. 2015.
[15] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. Int. Conf. Neural Netw. (ICNN), Perth, WA, Australia, vol. 4, 1995, pp. 1942–1948.
[16] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” J. Global Optim., vol. 11, no. 4, pp. 341–359, 1997.
[17] S. Saremí, S. Mirjalili, and A. Lewis, “Grasshopper optimisation algorithm: Theory and application,” Adv. Eng. Softw., vol. 105, pp. 30–47, Mar. 2017.
[18] C. Formatted G.-G. Wang, L. Guo, A. H. Gandomi, G.-S. Hao, and H. Wang, “Chaotic krill herd algorithm,” Inf. Sci., vol. 274, pp. 17–34, Aug. 2014.
[19] A. W. Mohamed and P. N. Suganthan, “Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation,” Soft Comput., vol. 22, no. 10, pp. 3215–3235, May 2018.
[20] C. Li, Z. Niu, Z. Song, B. Li, J. Fan, and P. X. Liu, “A double evolutionary learning moth-flame optimization for real-parameter global optimization problems,” IEEE Access, vol. 6, pp. 76700–76727, 2018.
[21] Z. Li, Y. Zhou, S. Zhang, and J. Song, “Lévy-flight moth-flame algorithm for function optimization and engineering design problems,” Math. Problems Eng., vol. 2016, Jul. 2016, Art. no. 1429390.
[22] A. Jafari, T. Khalili, E. Babaei, and A. Bidram, “A hybrid optimization technique using exchange market and genetic algorithms,” IEEE Access, vol. 8, pp. 2417–2427, 2020.
[23] X. Zhao, X. Zhang, Z. Cai, X. Tian, X. Wang, Y. Huang, H. Chen, and L. Hu, “Chaos enhanced grey wolf optimization wrapped ELM for diagnosis of paratou-poisoned patients,” Comput. Biol. Chem., vol. 78, pp. 481–490, Feb. 2019.
[24] Z. Yan, C. Li, Z. Song, L. Xiong, and C. Luo, “An improved brain storm optimization algorithm for estimating parameters of photovoltaic models,” IEEE Access, vol. 7, pp. 77629–77641, 2019.
[25] H. Wang, W. Wang, Z. Cui, X. Zhou, J. Zhao, and Y. Li, “A new dynamic firefly algorithm for demand estimation of water resources,” Inf. Sci., vol. 438, pp. 95–106, Apr. 2018.
[26] H. Wang, W. Wang, H. Sun, Z. Cui, S. Rahnamayan, and S. Zeng, “A new cuckoo search algorithm with hybrid strategies for flow shop scheduling problems,” Soft Comput., vol. 21, no. 15, pp. 4297–4307, Feb. 2016.
[27] J.-H. Yi, S. Deb, J. Dong, H. Alavi, and J.-H. Yi, “An improved NSGA-III algorithm with adaptive mutation operator for big data optimization problems,” Future Gener. Comput. Syst., vol. 88, pp. 571–585, Nov. 2018.
[28] J.-H. Yi, L.-N. Xing, G.-G. Wang, J. Dong, A. V. Vasilakos, A. H. Alavi, and L. Wang, “Behavior of crossover operators in NSGA-III for large-scale optimization problems,” Inf. Sci., vol. 509, pp. 470–487, Jan. 2020.
[29] H. Huang, N. He, and D. Xu, “Optimal design for passive power filters in hybrid power filter based on particle swarm optimization,” in Proc. IEEE Int. Conf. Automat. Logistics, Jinan, China, Aug. 2007, pp. 1468–1472.
[30] Y.-P. Chang and C. Low, “A new crossover operator in NSGA-III for large-scale optimization problems,” Expert Syst. Appl., vol. 35, no. 3, pp. 934–940, Oct. 2008.
[31] Z. Juan, G. Yi-nan, and Z. Shu-ying, “Optimal design of passive power filters of an asymmetrical system based on genetic algorithm,” Procedia Earth Planet. Sci., vol. 1, no. 1, pp. 1440–1447, Sep. 2009.
[32] CR Formatted N.-C. Yang and M.-D. Le, “Optimal design of passive power filters based on multi-objective bat algorithm and Pareto front,” Appl. Soft Comput., vol. 35, pp. 257–266, Oct. 2015.

[33] A. F. Zobaa, “Optimal multiobjective design of hybrid active power filters considering a distorted environment,” IEEE Trans. Ind. Electron., vol. 61, no. 1, pp. 107–114, Jan. 2014.

[34] A. K. Tiwari and S. P. Dubey, “Ant colony optimization based hybrid active power filter for harmonic compensation,” in Proc. Int. Conf. Electr. Electron., Optim. Techn. (ICEEOT), Chennai, India, Mar. 2016, pp. 777–782.

[35] R. Jayap. R. N. Patel, and D. D. Neema, “Design of optimized hybrid active power filter for electric arc furnace load,” in Proc. Recent Develop. Control, Autom. Power Eng. (RDCAPE), Noida, India, Oct. 2017, pp. 105–110.

[36] Y.-H. Jiang and D.-F. Liao, “Multi-objective optimal design for hybrid active power filter based on composite method of genetic algorithm and particle swarm optimization,” in Proc. Int. Conf. Artif. Intel. Comput. Intell., Shanghai, China, 2009, pp. 549–553.

[37] W. Zhao, A. Luo, X. Deng, K. Zhou, and J. Wu. “Parameter design for improving injection type hybrid active power filter performance in high power grid,” in Proc. IEEE 6th Int. Power Electron. Motion Control Conf., Wuhan, China, May 2009, pp. 1148–1154.

[38] CR Formatted Y. Jing, J. Chang, and S. Tian, “Multi-objective optimal design of hybrid active power filter,” in Proc. Int. Conf. Adv. Manuf. Technol. Ind. Appl. (AMITA), Shanghai, China, Nov. 2016, pp. 138–142.

[39] P. P. Biswas, P. N. Suganthan, and G. A. J. Amaratunga, “Minimizing harmonic distortion in power system with optimal design of active power filter using differential evolution,” Appl. Soft Comput., vol. 61, pp. 486–496, Dec. 2017.

[40] S. Mirjalili, “Dragonfly algorithm: A new heuristic optimization technique for solving single-objective, discrete, and multi-objective problems,” Neural Comput. Appl., vol. 27, no. 4, pp. 1053–1073, May 2016.

[41] A. I. Hammad, E. T. A. Samra, M. A. Al-Betar, R. M. Khalil, Z. Alasmer, and M. Kan, “A dragonfly algorithm for solving traveling salesman problem,” in Proc. 8th IEEE Int. Conf. Control Syst., Comput. Eng. (ICCSSCE), Penang, Malaysia, Nov. 2018, pp. 136–141.

[42] F. Lagos, B. Crawford, R. Soto, J. Lemus-Romani, G. Astorga, A. Salas-Fernandez, and J.-M. Rubio, “Bridge strengthening by conversion to tied-arch using dragonfly algorithm,” in Proc. 14th Iberian Conf. Inf. Syst. Technol. (CISTB), Coimbra, Portugal, Jun. 2019, pp. 1–5.

[43] J. Peng, Y. Ye, S. Chen, and C. Dong, “A novel chaotic dragonfly algorithm based on sine-cosine mechanism for optimization design,” in Proc. 2nd Int. Conf. Inf. Syst. Comput. Aided Edu. (ICISCAE), Dalian, China, Sep. 2019, pp. 185–188.

[44] J. Song and S. Li, “Elite opposition learning and exponential function steps-based dragonfly algorithm for global optimization,” in Proc. IEEE Int. Conf. Inf. Autom. (ICIA), Macau, China, Jul. 2017, pp. 1178–1183.

[45] S. K. S. Ranjini and S. Murugan, “Memory based hybrid dragonfly algorithm for numerical optimization problems,” Expert Syst. Appl., vol. 83, pp. 63–78, Oct. 2017.

[46] C. Shilaja and T. Arunprasath, “Internet of medical things-load optimization of power flow based on hybrid enhanced grey wolf optimization and dragonfly algorithm,” Future Gener. Comput. Syst., vol. 98, pp. 319–330, Sep. 2019.

[47] L. Xu, H. Jia, C. Lang, X. Peng, and K. Sun, “A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution,” IEEE Access, vol. 7, pp. 19502–19538, 2019.

[48] D. H. Wolpert and W. G. Macready, “No free lunch theorems for optimization,” IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 67–82, Apr. 1997.

[49] G.-G. Wang and Y. Tan, “Improving metaheuristic algorithms with information feedback models,” IEEE Trans. Cybern., vol. 49, no. 2, pp. 542–555, Feb. 2019.

[50] Z.-M. Gu and G.-G. Wang, “Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization,” Future Gener. Comput. Syst., vol. 107, pp. 49–69, Jun. 2020.

[51] Y. Zhang, G.-G. Wang, K. Li, W.-C. Yeh, M. Jian, and J. Dong, “Enhancing MOEA/D with information feedback models for large-scale many-objective optimization,” Inf. Sci., vol. 522, pp. 1–16, Jun. 2020.

[52] Z. Ren, A. Zhang, C. Wen, and Z. Feng, “A scatter learning particle swarm optimization algorithm for multimodal problems,” IEEE Trans. Cybern., vol. 44, no. 7, pp. 1127–1140, Jul. 2014.

[53] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, “Comprehensive learning particle swarm optimizer for global optimization of multimodal functions,” IEEE Trans. Evol. Comput., vol. 10, no. 3, pp. 281–295, Jun. 2006.

[54] S. Rahmani, A. Hamadi, and K. Al-Haddad, “A comprehensive analysis of hybrid active power filter for power quality enhancement,” in Proc. 38th Annu. Conf. IEEE Ind. Electron. Soc. (IECON), Montreal, QC, Canada, Oct. 2012, pp. 6258–6267.

[55] IEEE Recommended Practices and Requirements for Harmonic Control in Electrical Power Systems. IEEE Standard 519-2014, 2014.

[56] IEEE Standard Definitions for the Measurement of Electric Power Quantities Under Sinusoidal, Balanced, or Unbalanced Conditions, IEEE Standard 1949-2010, 2010.

[57] J. Ronkkonen, S. Kuukkonen, and K. V. Price, “Real-parameter optimization with differential evolution,” in Proc. IEEE Congr. Evol. Comput., vol. 1, Sep. 2005, pp. 506–513.

[58] P. Jimakouoswiti, H. Akagi, H. Fujita, and S. Ogasawara, “Implementation and performance of automatic gain adjustment in a shunt-active filter for harmonic damping throughout a power distribution system,” IEEE Trans. Power Electron., vol. 17, no. 3, pp. 438–447, May 2002.

[59] J. Derrac, S. García, D. Molina, and F. Herrera, “A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms,” Swarm Evol. Comput., vol. 1, no. 1, pp. 3–18, Mar. 2011.

WANXUAN DAI is currently pursuing the degree with the School of Information Engineering, Nanchang University, Nanchang, China. His current research interests include swarm particle optimization, genetic algorithms, and other computational intelligence techniques.

CHUNQUAN LI (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from Nanchang University, Nanchang, China, in 2002, 2007, and 2015, respectively.

Since 2002, he has been with the School of Information Engineering, Nanchang University, where he is currently an Associate Professor and a Young Scholar of Ganjiang River. He has published over 30 research articles. He is also a Visiting Professor with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada. His current interests include computing intelligence, haptics, virtual surgery simulation, robotics, and their applications to biomedical engineering.

ZHILING CUI is currently pursuing the B.Sc. degree with the School of Information Engineering, Nanchang University, Nanchang, China. His current research interests include teaching-learning-based optimization, moth-flame optimization, and other computational intelligence techniques.
YUFAN WU is currently pursuing the B.Sc. degree with the School of Information Engineering, Nanchang University, Nanchang, China. His current research interests include optimization problems, swarm optimization algorithms, and other computational intelligence techniques.

LEYINGYUE ZHANG is currently pursuing the B.Sc. degree with the School of Information Engineering, Nanchang University, Nanchang, China. Her current research interests include particle swarm optimization, moth-flame optimization, and other computational intelligence techniques.

JUNRU HUANG is currently pursuing the B.Sc. degree with the School of Information Engineering, Nanchang University, Nanchang, China. His current research interests include particle swarm optimization, genetic algorithms, and other computational intelligence techniques.

* * *