Applied Intelligence (2023) 53:14918–14960
https://doi.org/10.1007/s10489-022-04224-6

A novel high-level target navigation pigeon-inspired optimization for global optimization problems

Hanming Wang[1] · Jinghong Zhao[1]

Accepted: 27 September 2022 / Published online: 5 November 2022

Abstract

The canonical Pigeon-inspired optimization (PIO) possesses excellent local exploitation capability and can provide a very fast convergence speed, but it is also easily trapped in local optima especially when facing complex problems. In order to take full advantage of the superior local exploitation capability, and at the same time enhance the global exploration capability of PIO, we present a modified PIO called high-level target navigation PIO (HTNPIO). The HTNPIO includes three strategies, selective mutation strategy (SMS), levy-based map-compass strategy (LMS), and enhanced landmark strategy (ELS). In the strategies, two kinds of mutation strategies and a simple levy-flight operator are performed. What's more, an LMS-ELS probability is proposed to balance the exploration and exploitation. In order to test the performance of the proposed optimizer, HTNPIO is made comparisons with other 15 PIO and advanced heuristic algorithms on the IEEE CEC2017 benchmark problems and 5 real world optimization problems. Experimental results demonstrate that HTNPIO defeats all the competitors on the CEC2017 benchmark problems including the extraordinarily competitive LSHADE, and also exhibits extremely competitive performance in dealing with the real-world problems. Therefore, HTNPIO might be effective to provide promising solutions in various function and industrial optimization problems.

Keywords Pigeon-inspired optimization (PIO) · Global optimization · Differential evolution · Levy-flight model

1 Introduction

Complex parameter optimization problems require more intelligent and automated computations in various practical industrial applications [1, 5–7, 11, 23, 25, 45, 53, 59]. Traditional gradient methods require function information and are not suitable for solving non-convex optimization problems [76]. To work out multimodal and discontinuous optimization problems more effectively, researchers have presented a large number of heuristic methods. The most popular heuristic methods include evolutionary algorithms and swarm intelligent algorithms, such as genetic algorithm (GA) [15, 47, 66], differential evolution (DE) [14, 26, 51, 55], particle swarm optimization (PSO) [12, 39, 58, 69], ant colony optimization (ACO) [19, 28, 50], artificial bee colony (ABC) [17, 57], Cooperative Framework for Fireworks Algorithm(CoFFWA) [74, 78], Moth-flame optimization algorithm (MFO) [40, 48, 56, 67] and so on. Most heuristic methods are inspired by the evolutionary law or survival law of the the natural species, which have been proved to be able to effectively solve the complex problems.

Pigeon-inspired optimization (PIO) is a new swarm intelligent optimization method, which has attracted widespread attention since proposed by Duan and Qiao [20] in 2014 for solving air robots path planning problems. Simulating the homing behavior of pigeons, PIO utilized the map and compass operator and landmark operator to find the solutions of problems, and can provide a very fast convergence rate, so it makes good sense for problems that require fast convergence in a short time. One of the most popular applications of PIO is to solve unmanned aerial vehicle (UVA) problems [18, 22, 24, 31, 32, 52, 64, 68]. Besides, due to its simple concept and efficient search, PIO has also been successfully applied to solve various industrial problems

Jinghong Zhao are contributed equally to this work.

Please email the author (6110116146@email.ncu.edu.cn) for the source code.

✉ Jinghong Zhao
zm.wang868@mails.ccnu.edu.cn

Hanming Wang
6110116146@email.ncu.edu.cn

[1] School of Electrical Engineering, Naval University of Engineering, Jiefang Road, Wuhan, 430030, Hubei, China

such as the air robots path planning problems [20, 73], feature selection problems [2, 49], and other critical problems [3, 9, 21, 29, 37, 77, 79]. PIO has been demonstrated to provide faster convergence speed and more accurate solutions than PSO and DE in many optimization problems [2, 18, 31, 73].

However, in order to quickly converge, canonical PIO kept all pigeons learning from the global best position (Gbest) of the entire swarm and actively reduced the population later in the iteration, which resulted in severe premature convergence when fixing higher dimensional problems. In order to better balance the exploration and exploitation abilities, which have been proved as two crucial characteristics of an algorithm [38], various PIO variants have been developed in recent years. Li and Duan [34] proposed a Bloch Quantum-behaved Pigeon-inspired optimization (BQPIO) using Quantum Evolutionary Theory for continuous optimization problems. In this method, the locations of all the pigeons are encoded by the probability amplitudes of quantum bits, and the quantum rotation gates cause the motion of pigeons and then attain diverse searching. BQPIO effectively improves the search capabilities of the original PIO. However, even if quantum behavior improves the performance of BQPIO, the learning method of BQPIO is still single, and the information interaction between particles is still lacking. Inspired by Prey-Predator model, Duan et al [8, 31] presented a predator-prey pigeon-inspired optimization (PPPIO) using predators to remove the bad solutions with specific hunting rate for UAV three-dimensional path planning and UAV ALS longitudinal parameters tuning problems. PPPIO simulates the behavior of predator-prey in nature, treating less-adapted individuals as predators and other individuals as prey. Prey try to move away from the predator to avoid falling into a local optimum. However, this method cannot learn from the optimal value and is very dependent on the selection of parameters, which is less efficient. Sun et al [70] adopted a heterogeneous comprehensive learning strategy to effectively improve the performance of PIO. Two strategies, heterogeneous comprehensive learning strategy and group-divided strategy are employed. In the grouping strategy, two subgroups divide the search effort, one focuses on the exploitation and the other on exploration. In the heterogeneous comprehensive learning strategy, if the fitness of the entire population does not improve within a certain number of generations, the heterogeneous comprehensive learning strategy is executed, so that a specific dimension of a vector is learned from the corresponding dimension of another vector. HCLPIO enhances the comvergence speed of PIO and has been successful in optimizing the application of fractional-orders controllers. Nevertheless, the search strategy of PIO does not provide a good global search capability, so even if individuals learn among different dimensions, there is no more potential learning object. TAO and Li [27] proposed a cross pigeon-inspired optimization with cognitive factor (CPIO-C) that adds nonlinear incremental inertia weight to the velocity and crosses the map-compass operator and landmark operator. Specifically, perturbation coefficients based on sine and cosine are added to the particle's flight velocity, which can realize a larger step search in the early stage and a preciser search in the later stage compared with PIO. What's more, CPIO-C enables compass and landmark strategies to cross-run with each other, which increases the continuous optimization capability of PIO. Although the perturbation coefficient provides PIO with a larger potential search space and can improve the convergence accuracy to a certain extent, the perturbation coefficient depends on parameter settings, and the algorithm may have completely different performances for different applications. In addition, the learning strategy is limited, making it less likely to search for the optimal value. Zhang and Duan [75] presented a social-class pigeon-inspired optimization (SCPIO) to optimize the multi-UAV path planning problem. SCPIO divides the entire population into top-to-bottom classes, and individuals in the lower classes can only learn from the higher classes, thus achieving unidirectional communications. The disadvantage of this method is that information can only be transmitted in one direction, and the efficient communications of population information cannot be realized. Zhang and Duan [18] proposed a levy flight-based pigeon-inspired optimization (LFPIO), employing a Levy flight operator to improve the capability of random search. Levy flight model is applied to adjust the step length of pigeons in both map-compass operator and landmark operator. By this method, pigeons randomly search in the search space with different step length. LFPIO effectively improves the global search capability of PIO, but it is still prone to falling into local optima because the exemplar are not high-quality. [64] employed cauchy mutation method instead of uniform distribution to jump out of the local optimum, and competed PSO, DE and PIO on the UAV path planning problem. However, the learning mechanism is still single, causing CMPIO to be unable to jump out of the local optimum when faced with complex problems.

However, most of PIO variants still stuck in local optima easily especially when facing complex multimodal problems. Actually, this is attributed to such a fact that there are not sufficient interactions between superior pigeons, which makes it difficult to acquire useful information exchanges to explore more potential solutions. To enhance the interaction between individuals in the population and give full play to the performance of the algorithm itself, employing single or ensemble mutation strategies (MSs) is a recently trend method. [81] introduced MS "$DE/rand/1$" to the brain storm optimization (BSO) for applications of ANNs, which enhanced the performance of BSO. [10] adopted an ensemble MS including "$DE/current-to-lbest/1$", "$DE/lbest/1$", "$DE/lbest/2$" to produce better guidance for particles. [41] applied $DE/rand/1$ to moth-flame optimization (MFO), where the elite moths operate MS to generate flames, and then the flame guided the moths to explore the potentiality, thus establishing a positive feedback channel. It maintained the good exploitation performance and enhanced the global search capability of MFO. [43] introduced a differential mutation strategy to increase the population diversity and enhance the global exploration performance of particle swarm optimization (PSO), along with a multidimensional comprehensive learning strategy. Inspired by MS, [80] modified a solution search equation for the employed bee phase of the artificial bee colony algorithm (ABC). [76] applied a composite MS to salp swarm algorithm (SSA) to generate trial vectors. In CMSRSSSA, for each individual, three MSs "$DE/rand/1$", "$DE/rand/1/bin$" and "$DE/rand/2/bin$" are simultaneously executed to make a competition to select a superior one as a trial vector, which enhanced the exploration and exploitation of SSA. CMSRSSSA is demonstrated to outperform the winners of the related IEEE CEC competition.

The success of the above algorithms is jointly attributed to the inspiration of MS. However, since each algorithm has special properties, MSs should be combined with the specific algorithms in unique ways. Dialectically, finding a proper and efficient way to implement MSs can greatly enhance the performance of the specific algorithm, otherwise it will only waste computing resources. For example, PIO has the weakness that it is easy to fall into local optimum, so the MSs with more random search capabilities are much more suitable for PIO.

In this paper, a high-level target navigation pigeon-inspired optimization (HTNPIO) is presented with three strategies: selective mutation strategy (SMS), levy-based map-compass strategy (LMS), and enhanced landmark strategy (ELS). The selective mutation method ("$DE/rand/1/bin$" or "$DE/rand/2/bin$") is utilized to generate high-level targets to improve the exploration ability of PIO, and the levy-flight method is implemented in LMS. Moreover, the landmark operator is modified in HTNPIO, in order to take advantages of PIO's local exploitation capability. In addition, an LMS-ELS probability (LP) mechanism is proposed to maintain a trade off between exploration and exploitation. Visually, the overview diagram of HTNPIO is illustrated in Fig. 1, and the main contributions of this paper are summarized as follows:

1. Apply a selective mutation strategy (SMS) to allow substantial information interactions between different elite pigeons to explore more potential solutions. In SMS, $DE/rand/1/bin$ or $DE/rand/2/bin$, and crossover method is adopted to generate high-level exemplars (targets). Pigeons learn from high-level targets instead of single global best. Therefore, the exploration capability of PIO is enhanced.
2. A simple levy-flight operator is introduced in levy-based map-compass strategy (LMS) to enhance the random search capability of PIO.
3. In order to give full play to the original exploitation and fast converge ability of PIO, the landmark operator of PIO was modified and executed in each iteration crossing with LMS.
4. An LMS-ELS probability (LP) mechanism was proposed to control the execution frequency of the LMS and ELS in each generation. It effectively improves the balance between the exploration and exploitation.
5. The reinforced HTNPIO defeats all the competitors on CEC2017 benchmark problems including the extremely competitive LSHADE, and achieves a best-in-class performance in dealing with practical optimization problems, which suggests that HTNPIO has broad applicable prospects.

The remainder of the paper is arranged as follows. Section 2 presents the related work, introducing the canonical PIO, DE, and levy flight model. Section 3

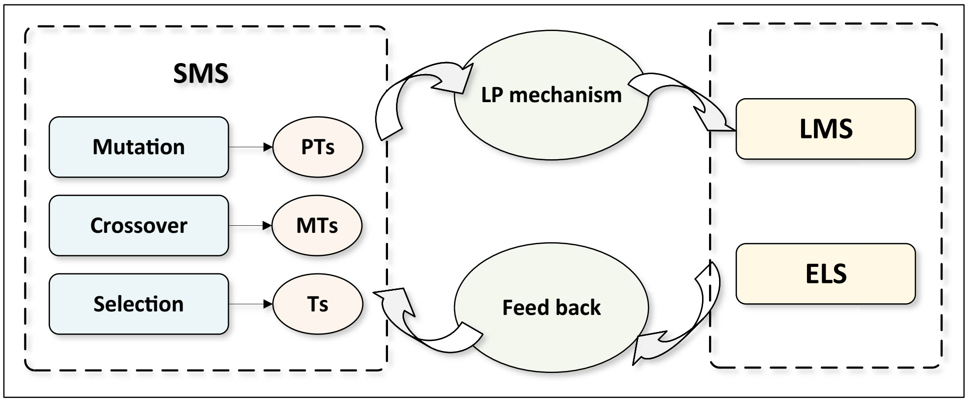


Fig. 1 Overview diagram of HTNPIO

demonstrates the proposed HTNPIO in detail. Section 4 analyzes the computational complexity of the presented HTNPIO. Section 5 visualize the performance of strategies of HTNPIO. The experimental results of 30 CEC2017 benchmark instances and 5 real world problems are given in Sections 6 and 7, respectively. Discussion is in Section 8. Conclusion is provided in Section 9.

2 Related works

2.1 Pigeon-inspired optimization

Enlightened by homing pigeons, Duan and Qiao first proposed PIO in 2014, which contains two operators: the map-compass operator and landmark operator. In the early stage of iterations, the map-compass operator is conducted to provide the exploration for the solutions; in the later stage, PIO utilizes landmark operator to accelerate convergence. The maximum iterative number of the two operators are $t_1^{\max}$ and $t_2^{\max}$, respectively. Note that the position of a pigeon represents a probable solution of the objective function, and its corresponding fitness value represents the objective function value. For instance, for minimum optimization problems, a solution with smaller fitness value represents a better position. Initially, we use formulas (1) and (2) to initialize the position and velocity of pigeons. $X_{i,d}$ and $V_{i,d}$ are the positions and velocity of the i-th pigeon in the d-th dimension, respectively. $B_{\min}$ and $B_{\max}$ are the minimum and maximum bound of $X_{i,d}$, respectively. "$rand$" indicates an evenly distributed random number in the range [0,1].

$$X_{i,d} = B_{\min} + rand \cdot (B_{\max} - B_{\min}) \tag{1}$$

$$V_{i,d} = rand \tag{2}$$

2.1.1 Map-compass operator

For the t-th generation, $t \in [1, t_1^{\max}]$, the map-compass operator is conducted. In this strategy, the velocity of pigeons are disturbed by an attenuation factor $e^{-R \cdot t}$, where the attenuation speed is determined by the map-compass factor R. The position and velocity of the i-th pigeon in the t-th generation are updated by the following formulations (3) and (4):

$$V_i(t+1) = V_i(t) \cdot e^{-R \cdot t} + rand \cdot [X_g - X_i(t)] \tag{3}$$

$$X_i\,(t+1) = X_i + V_i\,(t+1) \tag{4}$$

where X_g is the global best position of pigeons obtained so far, and $rand$ is a random number uniformly distributed between 0 and 1.

2.1.2 Landmark operator

When $t_1^{\max} < t \leq t_2^{\max}$, the landmark operator is implemented. In the landmark operator, the number of pigeons NP is consistently decreased in every generation based on formula (5) as follows:

$$NP\,(t+1) = \frac{NP(t)}{2} \tag{5}$$

The pigeons fly towards their destination by following landmarks in this strategy. If pigeons are familiar with the landmarks, they will directly fly to the destination. If they are away from the destination or unfamiliar with the landmarks, they will follow the pigeons who are familiar with the landmarks. X_c^t stands for the landmark and is the center of all pigeons position at the t-th iteration. The half of the pigeons that are away from the destination will follow the center pigeon. Thus, the pigeons can jointly fly to their destination quickly.

The position updating rule for the i-th pigeon at the t-th iteration can be given by:

$$X_c(t) = \frac{\sum_1^{NP(t)} X_i(t) f(x_i(t))}{NP(t) \sum_1^{NP(t)} f(x_i(t))} \tag{6}$$

$$f(x_i(t)) = \begin{cases} \frac{1}{p_i^t + \varepsilon}, forminimumproblems \\ p_i^t, formaximumproblems \end{cases} \tag{7}$$

$$X_i\,(t+1) = X_i^t + rand \cdot [X_c^t - X_i(t)] \tag{8}$$

Here, $f\left(x_i^t\right)$ is the quality of the pigeon's position, and p_i^t is the fitness value of the i-th pigeon in the t-th generation. $rand$ is a uniformly distributed number between 0 and 1. When $t > t_2^{\max}$, the iterative search for the PIO algorithm is end.

2.2 Differential evolution algorithm

Differential evolution (DE) algorithm presented by [55] is an efficient evolutionary computing method for working out various complex optimization problems. Three strategies, mutation, crossover and selection are proposed in DE algorithm. DE algorithm intends to evolve a population of N individuals in D-dimension search space towards the global optimum by applying the mutation, crossover, and selection strategy.

2.2.1 Mutation strategy

The mutation strategy utilizes existing individuals $X_i = \left[x_{i1}, x_{i2}, \cdots, x_{i,d}\right], i \in \{1, 2, \cdots, N\}$ to generate mutant vector $V_i = \left[v_{i1}, v_{i2}, \cdots, v_{i,d}\right], i \in \{1, 2, \cdots, N\}$. Usually, there are five differential evolution strategies [55].

Particularly, "$DE/rand/1/bin$" and "$DE/rand/2/bin$" are the used mutation strategies in this paper, described as follows, where $r1$, $r2$, $r3$, $r4$ and $r5$ are five incompatible integers randomly selected from $[1, N]$, which are also different from the ith individual, t is the number of generation.

$DE/rand/1/bin$:

$$\begin{aligned} Vector_i^t = & X_{r1}^t + F \cdot (X_{r2}^t - X_{r3}^t) \\ & + F \cdot (X_{r4}^t - X_{r5}^t) \end{aligned} \tag{9}$$

$DE/rand/2/bin$:

$$\begin{aligned} Vector_i^t = & X_i^t + F \cdot (X_{r1}^t - X_i^t) \\ & + F \cdot (X_{r2}^t - X_{r3}^t) \end{aligned} \tag{10}$$

here, F is the mutation operator that determines the magnification ratio.

2.2.2 Crossover strategy

The next binomial crossover strategy is utilized to produce a trial vector $U_i = \left[u_{i1}, u_{i2}, \cdots, u_{i,d}\right], i \in \{1,\ 2,\ \cdots,\ N\}$ and the trial vector can either be the original vector X_i or its mutant vector V_i depending on the following formulation (10), where i is the individual serial number, and d is the dimension serial number:

$$u_{i,d} = \begin{cases} vector_{i,d}, (rand \le CR)\ or\ d = d_{rand} \\ x_{i,d}, (rand > CR)\ or\ d \ne d_{rand} \end{cases} \tag{11}$$

where $u_{i,d}$ is the i-th trial vector at the d-th dimension, $rand$ is a decimal uniformly generated from [0, 1], and $CR \in [0, 1]$ is the crossover rate. d_{rand} is the number randomly chosen from 1 to D, ensuring that U_i is not same as X_i in the d-th dimension.

2.2.3 Selection strategy

In the selection strategy, U_i and X_i are compared by its fitness value to select the better one as the X_i in next generation, i is the current number of the individual, and $t+1$ indicates the next generation. $X_i\ (t+1)$ is updated by (12) for the minimum problems.

$$X_i\ (t+1) = \begin{cases} U_i(t), & f[U_i(t)] < f[Xi(t)] \\ X_i^t, & f[U_i(t)] \ge f[Xi(t)] \end{cases} \tag{12}$$

where $f[U_i(t)]$ and $f[X_i(t)]$ represent the fitness value of $U_i(t)$ and $X_i(t)$, respectively.

2.3 Levy-flight model

Recent diverse heuristic algorithms have exhibited the advantages of Levy distribution [30, 33, 35, 36, 44, 54, 63], which models a random walk with frequent small steps and occasional big steps. Because the probability density of the step sizes of levy-flight is heavy-tailed, the characteristic allows the search to jump out of a local optimum and start with a new region in search space. Typically, a levy-flight based motion pattern goes like: a particle moves locally, walking small steps frequently, and performs big steps occasionally, and then moves locally again. The mathematical model of the levy flight distribution is defined as follows:

$$\chi = \frac{\lambda}{\|\theta\|^{\frac{1}{\eta}}}, \quad \lambda \sim N(\mu, \sigma_\lambda^2), \quad \theta \sim N(\mu, \sigma_\theta^2) \tag{13}$$

$$\sigma_\lambda = \left[\frac{\Gamma(1+\eta) \cdot sin(\frac{\pi\eta}{2})}{2^{\frac{\eta-1}{2}} \cdot \Gamma(\frac{1+\eta}{2}) \cdot \eta}\right]^{\frac{1}{\eta}} \tag{14}$$

In this paper, the decision parameters are as follows:

$$\mu = 0, \quad \sigma_\theta = 1, \quad \eta = 0.1 \tag{15}$$

3 High-level target navigation pigeon-inspired optimization

The basic PIO obtains the optimum by following the current leader and updating the followers. If the leader is fallen into local optima, the followers are likely to stuck in local optimum. Therefore, the performance of PIO depends largely on the quality of exemplars. To improve the quality of the exemplars while maintaining the exploitation ability of PIO, HTNPIO is presented with three strategies, selective mutation strategy (SMS), levy-based map-compass strategy (LMS), and enhanced landmark strategy (ELS). First two strategies can make different pigeons obtain frequent interactions for increasing the capacity of global search. The third strategy commits to accelerate convergence. In SMS, we apply two alternative mutation strategies, crossover, and selection operations for the random personal best pigeons (PBs) to build high-level targets. In LMS, the target can be employed to guide the pigeons for exploring more promising solutions, while levy-flight model is introduced to enhance the randomness. In ELS, landmark operator of PIO is operated iteratively to enhance the exploitation. What's more, a linear LMS-ELS probability (LP) mechanism is proposed to balance the exploration and exploitation. Owing to the navigation of the targets, the pigeons of finding better solutions can be evolved into the new PBs. Such PBs can be further used to produce higher-level targets. In this way, targets and PBs could efficiently promote each other so as to build a positive feedback between them. The details of the HINPIO are given as follows.

3.1 Initialization

In initialization phase, the positions and velocities of pigeons are initialized as same as (1) and (2). Moreover, the targets are initialized as follows, specificly, indices ini represents the initial generation, and d is the number of the current dimension:

$$T_{i,d}^{ini} = B_{\min} + rand \cdot (B_{\max} - B_{\min}) \tag{16}$$

where $B_{\max}$ and $B_{\min}$ are maximum value of bound and minimum value of bound, respectively.

3.2 Selective mutation strategy (SMS)

Inspired by DE algorithm, we adopted mutation, crossover, and selection mechanism to establish connection between different PBs to build the guiding targets. The targets are random, preferable, and rapidly changed to guarantee the diversity and quality of the population. Targets can be divided into three ranks, namely primary targets (PTs), intermediate targets (MTs), and high-level targets (Ts). PTs are first generated by mutation strategy, and evolve into MTs after crossover. Finally, MTs are filtered to become the Ts. Three steps of construction are explained in details as follows.

3.2.1 Mutation

Mutation strategy is first utilized to generate primary targets (PTs). Two mutation strategies "$DE/rand/1/bin$" and "$DE/rand/2/bin$" can be randomly selected to generate PTs by a mutation factor ψ for each individual. What's more, the vector used to generate the target must be superior because we cannot generate targets from mediocrities because we need effective communications to economize time and space when facing complexities. To balance performance and costs, we choose personal best positions (PBs) of each pigeon $PB_i = [PB_{i1}, PB_{i2}, \ldots, PB_{i,d}], i \in [1, 2, \ldots, NP]$ as the basis to implement mutation strategy, where i is the current pigeon number, d is the current number of dimension of the pigeon, and NP is the total number of the pigeons. In initialization phase, we let every pigeon be a PB, with generations increasing, only the pigeons with better positions can evolve into personal best ones. The primary targets are generated by the following selective equation:

$$\begin{aligned} PT_i^t = {} & PB_{r1}^t + F_1 \cdot (PB_{r2}^t - PB_{r3}^t) \\ & + F_2 \cdot (PB_{r4}^t - PB_{r5}^t), c_i(t) > \psi \end{aligned} \tag{17}$$

$$\begin{aligned} PT_i^t = {} & PB_i^t + F_3 \cdot (PB_{r1}^t - PB_i^t) \\ & + F_4 \cdot (PB_{r2}^t - PB_{r3}^t), \quad c_i(t) < \psi \end{aligned} \tag{18}$$

where $r1, r2, r3, r4, r5$ are different numbers in the range $[1, NP]$, which are also not the same as i. PB_{r1-r5} denote five personal best pigeons that randomly selected from the entire pigeon swarm. $c_i(t)$ is a uniform random number in $[0, 1]$ for individual i in the t-th generation. $F_k, k = [1, 2, 3, 4]$ is the mutation operator that determines the magnification ratio between the vectors. In HTNPIO, F_1, F_2 and F_3 are taken as different uniformly distributed random number for each individual i to balance the exploration and exploitation, in this paper, F_4 is preset as 1.

Additionally, after the mutation, a boundary process method is introduced for each dimension of a PT who searches out of the search sapce as (19). In (19), t is the current iteration number, and d is the current dimension, and $B_{\max}$ and $B_{\min}$ are maximum and minimum value of the boundary, respectively.

$$PT_{i,d}^t = B_{\min} + rand \cdot (B_{\max} - B_{\min}) \tag{19}$$

Note that in the i-th PT vector, only the variables that search out of boundary perform (19).

3.2.2 Crossover

Intermediate targets (MT) are selected from PBs and PTs by the binomial crossover strategy as (21). In (21), $i \in [1, 2, \ldots, NP]$ denotes the current pigeon number, d is the current dimension number, and d_{rand} is the number randomly chosen from 1 to D.

$$MT_{i,d} = \begin{cases} PB_{i,d}, (rand \leq CR) & or \quad d = d_{rand} \\ PT_{i,d}, (rand > CR) & and \quad d \neq d_{rand} \end{cases} \tag{20}$$

where $MT_i = [MT_{i,1}, MT_{i,2}, \ldots, MT_{i,d}], i \in [1, 2, \ldots, NP]$

$$CR = \begin{cases} 0.5 \times (1 + rand), c_i(t) > \psi \\ \quad 0.9 \qquad\quad, c_i(t) < \psi \end{cases} . \tag{21}$$

where rand is a uniformly number in the range [0, 1]. As is seen, for for different mutation strategy, value of CR is different. For (17) and (18), CR is set as (19) correspondingly. This can guarantee each mutation method is utilized efficiently, which could maintain the diversity of the pigeon group.

3.2.3 Selection

Selection is the last step of the first strategy, which aims to filter out high-level targets from the intermediate targets. The target group is also a dynamic population, old targets must be substituted by better ones. Selection part executes the following functions to filter out the final targets.

$$T_i^{t+1} = \begin{cases} T_i^t, & f(T_i^t) < f(MT_i^t) \\ MT_i^t, & f(T_i^t) \geq f(MT_i^t) \end{cases} \tag{22}$$

where minimization problems are considered, $f(T_i^t)$ is the fitness value of the i-th target in the t-th generation, and $f(MT_i^t)$ is the fitness value of the i-th intermediate target in the t-th generation. The smaller fitness value indicates the closer distance to the global best position so that this position will be selected as the new target for the next generation. T_i^t is the i-th target in the $(t+1)$-th generation. Selection strategy can make sure that pigeons will always follow the targets with higher performance.

3.3 Levy-based map-compass strategy (LMS)

In levy-based map-compass strategy (LMS), pigeons fly towards the targets and explore potential positions around the targets. LMS is modified based on the original map-compass strategy of PIO. Instead of a stable value $R = 0.2$ in PIO, the map-compass factor $R_{i,d}^t$ in HTNPIO is set as a uniform random number in $[0, 0.25]$ for each variable d of i-th pigeon, and varies with iteration t. Second, the levy-flight model is introduced to LMS to enhance the random search ability. Considering NP homing pigeons searching within in a D-dimensional search space, the i-th pigeon's d-th ($d \in [1, D]$) dimensional position and velocity in the t-th generation are updated according to (23) and (24):

$$V_{i,d}^{t+1} = V_{i,d}^t + rand \cdot (T_{i,d}^t - X_{i,d}^t) \tag{23}$$

$$X_{i,d}^{t+1} = T_{i,d}^t \times (1 - e^{-R_{i,d}^t \cdot t}) + \chi \cdot V_{i,d}^{t+1} \tag{24}$$

where $V_i = [V_{i,1}, V_{i,2}, \ldots, V_{i,d}], i \in [1, 2, \ldots, NP]$ is the velocity of the ith pigeon, and $X_i = [X_{i,1}, X_{i,2}, \ldots, X_{i,d}], i \in [1, 2, \ldots, NP]$ indicates the ith pigeon's position within the D-dimension search space, and $d \in [1, D]$. $T_i = [T_{i1}, T_{i2}, \ldots, T_{i,d}], i \in [1, 2, \ldots, NP]$, is the ith high-level target. $R_{i,d}^t$ is the map-compass factor for the dth variable of the ith pigeon in t-th generation. Furthermore, χ is the levy-flight step length.

The concept map of LMS is shown in Fig. 2, where $Pbest_i$ is the personal best position of the i-th pigeon X_i, $i \in [1, NP]$ from the entire pigeon swarm. SMS can use different $Pbests$ to construct a high-level target; then, from the concept map, the pigeon X can fly towards the target position to explore better positions around this target via LMS. Due to the two strategies, the pigeon may explore a new potential search space in this iterative process.

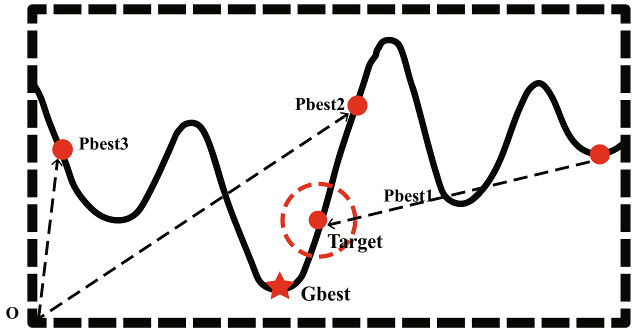


Fig. 2 Schematic diagram of the LMS

3.4 Enhanced landmark strategy (ELS)

PIO utilized map-compass strategy to enhance local exploitation capabilities and accelerate convergence in the later search stage. In original map-compass strategy of PIO, center of the current population is treated as the landmark, all the pigeons move towards the landmark, and the population is reduced with iteration. This method is efficient to exploit accurate solutions, but it reduces population diversity. To address the problem, in ELS, we no longer use the center of the entire population as a landmark, but select some elites of the targets and use their center as a landmark to navigate the movement of the pigeons. Specifically, the targets are divided into n groups with each group m pigeons, then, pick an elite from each group according the fitness value, and the center position of these elites is calculated as C_e^t, where t is the current iteration number. The selection of elite targets in groups is performed in every generation, and only when a target defeats the elite of the current group, can it become the new elite of this group. Note that the group is never changed. What's more, we make ELS and LMS cross in each generation, instead of only executing ELS in later iterations. In this way, the exploration capabilities of PIO is effectively retained. Additionally, in ELS, pigeons not only follow C_e, but also adjust their routes according to the corresponding targets.

In ELS, i indicates the current pigeon number, and d is the current dimension, the updating rule of the velocity $V_{i,d}$ and position $X_{i,d}$ is as follows:

$$V_{i,d}^t = V_{i,d}^{t-1} + rand \cdot \left(T_{i,d}^t - X_{i,d}^{t-1}\right) + rand \cdot (C_e^t - X_{i,d}^{t-1}) \tag{25}$$

$$X_{i,d}^t = C_e^t + V_{i,d}^t \tag{26}$$

Notably, for LMS and ELS, the boundary process is not the same as that of primary targets. If a pigeon searches out

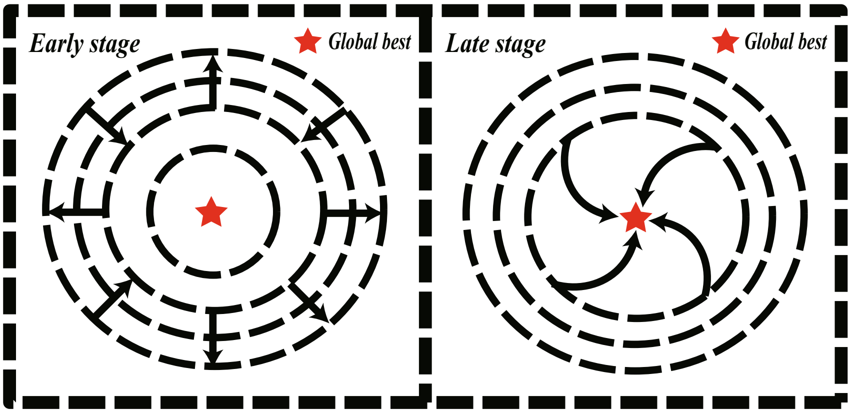


Fig. 3 Main idea of HTNPIO

of boundary, it will be pulled back to the upper or lower border as follows:

$$X_{i,out}^{t} = \begin{cases} B_{\max}, & X_{i,out}^{t} > B_{\max} \\ B_{\min}, & X_{i,out}^{t} < B_{\min} \end{cases} \tag{27}$$

3.5 LMS-ELS probability mechanism

Further, we propose an LMS-ELS probability (LP) mechanism in HTNPIO to balance the exploration and exploitation. The LP mechanism is a linearly incremental probability to control the execution number of the LMS and ELS in each generation. The LP mechanism is described as follows:

$$If\ rand_{i}^{t} > LP \begin{cases} yes, & execute \quad LMS \\ no, & execute \quad ELS \end{cases} \tag{28}$$

The linearly incremental LP is calculated by:

$$LP = \frac{t}{t_{\max}}, AP \in [0, 1] \tag{29}$$

where $i \in [1, NP]$, and $rand_{i}^{t}$ denotes a uniformly distributed number in [0, 1] for the i-th pigeon in the t-th generation, t and $t_{\max}$ mean the current generation and the maximum generation, respectively.

In the early iterative process, the probability value is far less than 1, so LMS has more execution opportunities to provide the sufficient exploration in the entire solution; conversely, in the lately process, the probability value is extremely high to focus on ELS for accelerating the convergence speed. Because of the LP probability, the proposed HTNPIO can achieve the efficient compromise between exploration and exploitation. Additionally, the main idea of the proposed HTNPIO is shown in Fig. 3. The figure illustrates that pigeons can explore as many potential areas as possible in the fore-stage iteration, while in the later stage, pigeons gathered to exploit the current optimal positions.

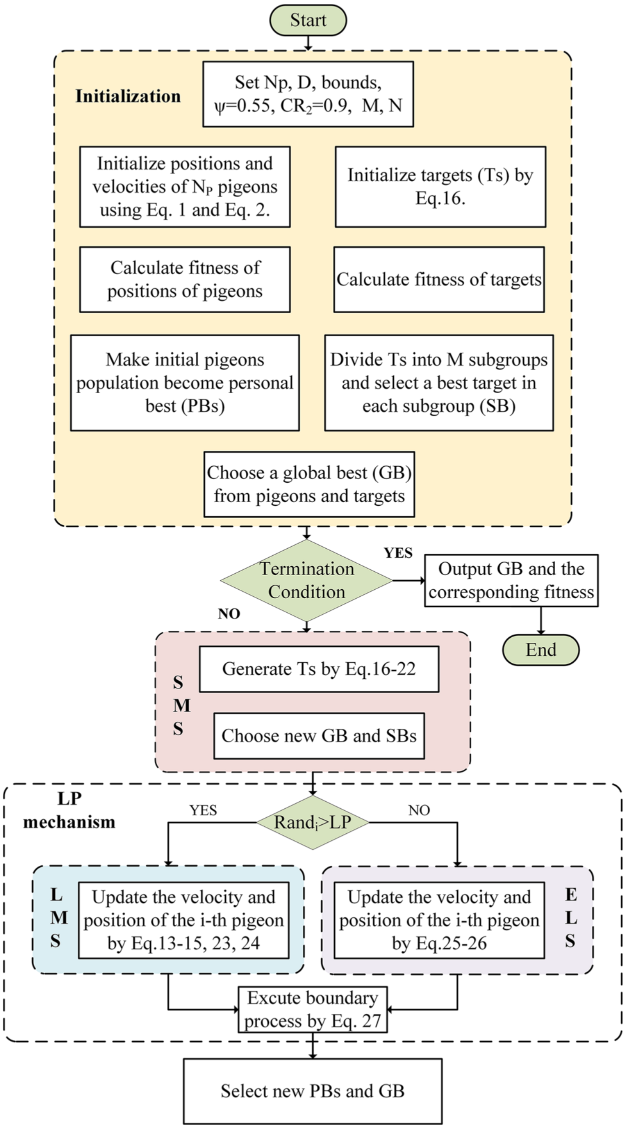

Start
Initialization
Set Np, D, bounds, ψ=0.55, CR₂=0.9, M, N
Initialize positions and velocities of N_P pigeons using Eq. 1 and Eq. 2.
Initialize targets (Ts) by Eq.16.
Calculate fitness of positions of pigeons
Calculate fitness of targets
Make initial pigeons population become personal best (PBs)
Divide Ts into M subgroups and select a best target in each subgroup (SB)
Choose a global best (GB) from pigeons and targets
Termination Condition
YES
Output GB and the corresponding fitness
End
NO
SMS
Generate Ts by Eq.16-22
Choose new GB and SBs
LP mechanism
YES
Rand_i>LP
NO
LMS
Update the velocity and position of the i-th pigeon by Eq.13-15, 23, 24
Update the velocity and position of the i-th pigeon by Eq.25-26
ELS
Excute boundary process by Eq. 27
Select new PBs and GB

Fig. 4 Flowchart of the complete HTNPIO

3.6 Procedure of HTNPIO

The complete flowchart of HTNPIO is in Fig. 4. The corresponding Pseudo code is given in Algorithm 1. The pseudo code of HTNPIO is completely showed in the chart. In Algorithm 1, $f[]$ means the objective function value while Gbest represents the global best position. And the value of CR is reset at the beginning of every generation to promise the capacity of global search.

Algorithm 1 HTNPIO.

```
%Initialization
Set number of pigeons ($N_P$), problem dimension ($D$), boundary values, $CR_1^{ini} = 0.5$, $\psi = 0.55$, $CR_2 = 0.9$, number of subswarms ($M$), number of pigeons per subswarm ($N$).
Initialize positions and velocities of NP pigeons using (1) and (2).
Evaluate corresponding fitness values of pigeons.
Let initial population beccome personal best pigeons ($PBs$) and choose the global best pigeon ($GB$).
Initialize targets ($Ts$) by (16);
Divide all $Ts$ into $M$ subgroups and select the best individual in each subgroup ($SBs$).
while (termination criteria is not arrived) do
$CR = 0.5 \times (1 + rand)$;
$t = t + 1$;
%SMS
for i=1 to $NP$ do
    Generate targets ($Ts$) using (16).(22);
    Choose new SBs and GB according to Ts' fitness.
    $R = 0.25 \cdot rand(1, D)$;    %Map-compass operator
    if $rand > LP$ then
        Calculate Levy-flight step length by (13)-(15);
        %LMS
        Update the $i$-th velocity by (23);
        Update the $i$-th position by (24);
    else
        %ELS
        Update the $i$-th velocity by (25);
        Update the $i$-th position by (26);
    end if
    All out-of-bounds variables are processed according to (27);
    Select personal best pigeons ($PBs$) and the global best pigeon ($Gbest$);
end for
end while
```

Algorithm 1 HTNPIO.

4 The computational complexity analysis

For decreasing computational cost when resolving practical optimization issues, it is hoped that the computational complexity of the algorithms are as low as possible. In this section, the computational complexity of HTNPIO will be calculated step by step.

First, number of the population (NP), dimension of the functions (D), and number of iterations ($T/2$) should be reaffirmed for the complexity calculation of HTNPIO. It should be noted that with the same maximum number of function evaluations, HTNPIO evaluates both targets and pigeons in an iteration, so the iteration number of HTNPIO is half of that of PIO. Therefore, the number of iterations of PIO is T.

In the initial phase, HTNPIO initializes both targets and pigeons, thus, the complexity is $O(2NP \times D)$, which is as twice as that of classical PIO.

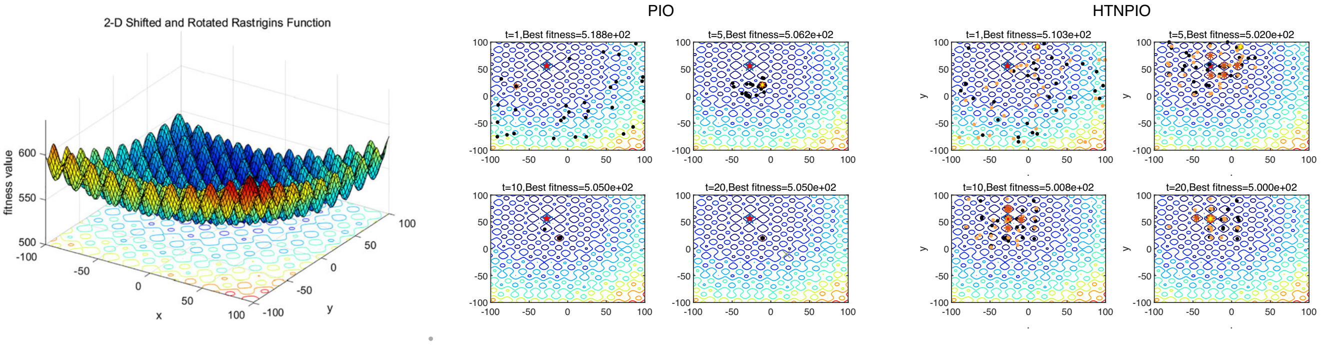
In the main loop of HTNPIO, SMS selects (17) or (18) to generate primary targets, and adopts crossover and selection strategy to produce high-level targets. The computational complexity of selected mutation strategy, crossover, and selection is computed as $O\left[(T-1)/2 \times NP \times D\right]$, $O\left[(T-1)/2 \times NP \times D\right]$, $O\left[(T-1)/2 \times NP \times D\right]$, respectively. Therefore, the computational complexity of SMS is $O\left[3(T-1)/2 \times NP \times d\right]$. Further, ELS or LMS is selected by LP mechanism to update the location of pigeons, thus the computational complexity is $O\left[(T-1)/2 \times NP \times D\right]$.

Therefore, the total computational complexity of HTNPIO is $O(2 \times T \times NP \times D)$. The computational complexity of conventional PIO can be simplified as $O\left(T \times NP \times D\right)$. It can be concluded that the computational complexity of HTNPIO is with the same order of magnitudethat as that of PIO. Considering that when solving real-world optimization issues, the computational cost is rather high, so such extra time cost can be acceptable.

5 Visual simulation of the performance of strategies of HTNPIO

In order to prove the effectiveness of HTNPIO's strategies, we visualize the search behavior of HTNPIO on several 2D function graphs, three $2-D$ functions are chosen from CEC2017 test suite. The three functions are the $2-D$ shifted and rotated rastrigin's function, $2-D$ shifted and rotated schaffer's function, given by

$$z(x, y) = \sum_{i=1}^{D}(x_i^2 - 10cos(2\pi M(x_i - o_i^5)) + 10) \qquad (30)$$



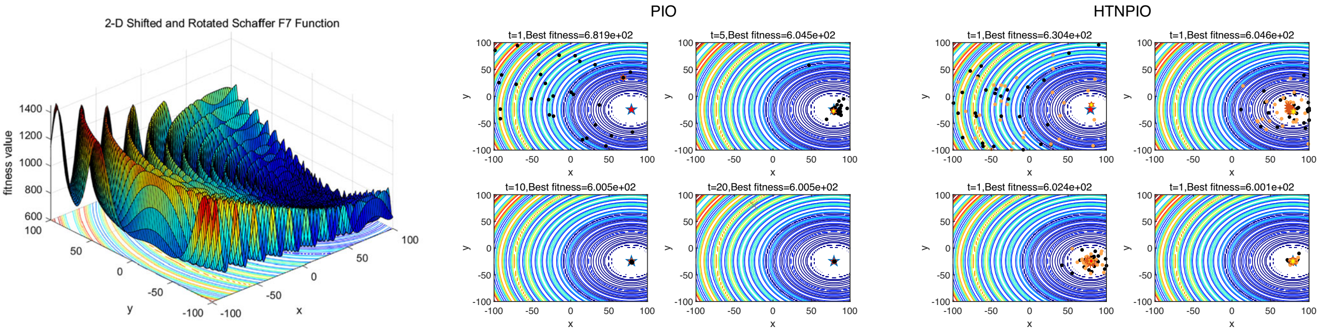
(a) (b) (c)

Fig. 5 (a) The 3-D surface and contour distribution of the 2-D multimodal rastrigin's function. (b) Contour map of the progressive searching performance of PIO. (c) Contour map of the progressive searching performance of the proposed HTNPIO

Figure 5(a) shows the three-dimensional graphics and contour distribution of the $2-D$ shifted and rotated Rastrigin's function, which is the 5th function of CEC2017. It is obvious that there are lots of local optima around the global minimum ($x = -27, y = 56, z = 500$), making searching groups easily get stuck in the local optima. Figure 5(b) and (c) separately demonstrate the iterative progression step by step performed by conventional PIO and the proposed HTNPIO with thirty individuals. The maximum iteration is set to 20, and the performance of the algorithms will be dynamically displayed at the 1-th, 5-th, 10-th, and 20-th iteration.

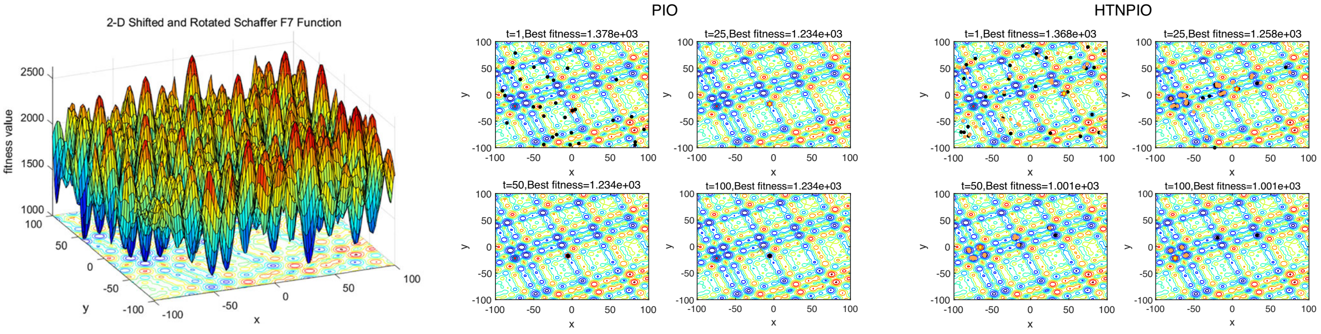
Figure 5(b) shows the convergence behavior of the conventional PIO. In the figure, the red five-pointed star is the preset global optima. The yellow six-pointed star is the obtained global optima. The black spots represent the personal best pigeons, and the green X is the center pigeon. In PIO, the global best position is found around $(-10, 25)$ and has been trapped into local optima. This is because the monotonous global optimal learning strategy leads to the deficiency of the population diversity, therefore cannot effectively guide pigeons to jump out of the local optima.

Figure 5(c) visualizes the searching performance of the proposed HTNPIO. In Fig. 5(c), the red five-pointed star indicates the preset global optima, and the yellow six-pointed star is the obtained global optima. The black spots are the personal best pigeons, the orange spots represent the high-level targets, and the orange star are the head pigeons. We can see that in HTNPIO, the construction and the dispersion strategies make the targets and the pigeons decentralized throughout the iteration. Even though the global best position is obtained, the high-level targets can still navigate pigeons to explore potential places and create population diversity in the later iteration. As can be seen in the figure, at the 1-th iteration, the global best pigeon is far from the global optima ($x = -27, y = 56$). However, at around 5-th iteration, several high-level targets are generated near the global optima, and successfully navigate pigeons to explore the global optima. What's more, even the ELS allows more pigeons to learn from the elites, the population diversity is still retained thanks to the instruction of the SMS and the LMS. Ultimately, by comparison, we can see that the three strategies we proposed in HTNPIO are effective in improving the global search capability.



(a) (b) (c)

Fig. 6 (a) The 3-D surface and contour distribution of the 2-D multimodal Scaffer's F7 function. (b) Contour map of the progressive searching performance of PIO. (c) Contour map of the progressive searching performance of the proposed HTNPIO



(a) (b) (c)

Fig. 7 (a) The 3-D surface and contour distribution of the 2-D multimodal Schwefel's function. (b) Contour map of the progressive searching performance of PIO. (c) Contour map of the progressive searching performance of the proposed HTNPIO

Figure 6(a) displays the three-dimensional graphics and contour distribution of the $2-D$ shifted and rotated Schaffer's F7 function, which is the $6-th$ test function of CEC2017. The global optimal of the function is set around (73, −27). Figure 6(b) and (c) visualizes the searching performance of conventional PIO and the proposed HTNPIO, respectively.

In Fig. 6(b), the conventional PIO approached the global best position at the $10-th$ iteration, however, because the local optimal points become complicated near the global optima, the conventional PIO can not accurately find the global best position. The explanation of the phenomenon is the decreasing of the population diversity, therefore PIO is trapped into local optima. Differently, the Fig. 6(c) shows that the proposed HTNPIO successfully searched the global best position within 20 iterations. The same phenomenon can be found in Fig. 7. This is attributed to the appropriate combination of the proposed SMS, LMS and ELS, enhancing exploration and exploitation ability.

6 Experimental results on CEC2017 benchmark

The CEC2017 benchmark suite [4] is a proven effective test set for evaluating single swarm intelligent algorithms. This test set contains 30 enormously complex shifted rotated instances including unimodal, multimodal, rotated and shift rotated hybrid and composition problems. The search range of each function is in the range [-100,100]. Table 1 introduces the information of the 30 functions in CEC2017 benchmark.

In order to further evaluate of the performance of HTNPIO, five state-of-the-art PIO variants including conventional PIO [20], BQPIO [34], HCLPIO [70], CPIO-C [27], and LFPIO [18] and other 10 algorithms with excellent performance involving DEBSO [81], HCLPSO [46] , PSO-DLS [71], SADE [51], LSHADE [60], MPEDE [26], DELMFO [41], HCLDMSPSO [65], CMSRSSSA [76], and EESHHO [42] are fairly run to make comparisons on the CEC2017 test benchmarks with $30-D$ and $50-D$ problems. For all algorithms, the maximum number of the fitness evaluations (maxFEs) is given as $10000 \times D$. The population size of the applied algorithms is 30. What's more, the maximum number of iterations is assigned to the maxFEs divided by the population size. Therefore, for the aforesaid algorithms excluding LFPIO, HTNPIO and DELMFO, their maximum iterative search number is set to $10000 \times D/30$. Because HTNPIO executes the evaluation for both each pigeon and its corresponding high-level target in each iteration, the maximum iterative number is set to $10000D/60$. Similarly, for DELMFO, each moth and its corresponding flame are both evaluated in a generation, so the maximum number iteration of DELMFO is also set to $10000D/60$. As for LFPIO, the individuals are updated twice in an iteration, therefore, the maximum iteration number of LFPIO is also calculated as $10000D/60$.

6.1 Indicators of comparisons

The error mean (Mean) and standard deviation value (STD) are utilized as evaluation indicators for the above compared algorithms. They are defined as follows:

$$Error = f(Gbest) - optima \tag{31}$$

$$Mean = \frac{\sum_1^{runs} Error}{runs} \tag{32}$$

$$STD = \sqrt{\frac{sum_1^{runs}[f(Gbest) - mean]^2}{runs - 1}} \tag{33}$$

here, error is the difference between the searched global best fitness value and the optima of the function. Mean refers to the average value of the errors with the multiple running number on each CEC2017 benchmark function while STD

Table 1 CEC2017 Benchmark instances with search range: [-100, 100]

Types	No.	Instances	
Unimodal	f1	Shifted and Rotated Bent Cigar Instance	100
	f2	Shifted and Rotated Sum of Different Power Instance	200
	f3	Shifted and Rotated Zakharov Instance	300
Multimodal	f4	Shifted and Rotated Rosenbrock's Instance	400
	f5	Shifted and Rotated Rastrigin's Instance	500
	f6	Shifted and Rotated Expanded Scaffer's F6 Instance	600
	f7	Shifted and Rotated Lunacek Bi_Rastrigin Instance	700
	f8	Shifted and Rotated Non-Continuous Rastrigin's Instance	800
	f9	Shifted and Rotated Levy Instance	900
	f10	Shifted and Rotated Schwefel's Instance	1000
Hybrid	f11	Hybrid Instance 1 (N=3)	1100
	f12	Hybrid Instance 2 (N=3)	1200
	f13	Hybrid Instance 3 (N=3)	1300
	f14	Hybrid Instance 4 (N=4)	1400
	f15	Hybrid Instance 5 (N=4)	1500
	f16	Hybrid Instance 6 (N=4)	1600
	f17	Hybrid Instance 7 (N=5)	1700
	f18	Hybrid Instance 8 (N=5)	1800
	f19	Hybrid Instance 9 (N=5)	1900
	f20	Hybrid Instance 10 (N=6)	2000
Composition	f21	Composition Instance 1 (N=3)	2100
	f22	Composition Instance 2 (N=3)	2200
	f23	Composition Instance 3 (N=4)	2300
	f24	Composition Instance 4 (N=4)	2400
	f25	Composition Instance 5 (N=5)	2500
	f26	Composition Instance 6 (N=5)	2600
	f27	Composition Instance 7 (N=6)	2700
	f28	Composition Instance 8 (N=6)	2800
	f29	Composition Instance 9 (N=3)	2900
	f30	Composition Instance 10 (N=3)	3000

denotes the standard deviation. The number of runs is set to 30.

Moreover, we conducted the Wilcoxon signed-rank test [16] with a significance level of 0.05 to compare the proposed method with each of the other methods. If the p-value on the Wilcoxon signed-rank test is less than 0.05, a significant difference between the two comparative algorithms will occur. Otherwise, there is no significant difference between them.

6.2 Parameters setting

Table 2 displays the main parameters of each algorithm from the literature. Most of the parameters come from literature recommendations, and a few are adjusted for better performance. For example, in the CPIO-C method, instead of $a = 0.7$ and $b = 0.1$, we use $a = b = 0.5$. This is because $a = b = 0.5$ can provide better performance for CPIO-C on the CEC2017 benchmark. Furthermore, [A$\sim$B] represents that A linearly decreases or increases to B with the increasing of generation. Precisely, [A$\sim$B] denotes two possible cases. If A>B, A will dynamically decrease to B with the number of the iteration increasing; otherwise, A will increase to B.

6.3 Comparison with PIO algorithms

The proposed HTNPIO algorithm is made comparisons with five PIO algorithms (PIO, BQPIO, CPIO-C, HCLPIO, LFPIO) on the 30 CEC2017 problems with 30-D,50-D in this section. For each indicator of an instance in Tables 3 and 4, the best result is bolded. For the six PIO algorithms, their population size and the maximum number of the fitness evaluations (FEs) are uniformly set as mentioined

Table 2 Pivotal Parameter Settings within 100-D in the range [-100,100]

Algorithms	Year	Parameter Settings	Ref.
PIO	2014	$R = 0.3$	[20]
BQPIO	2014	$R = 0.2, m = [1 \sim 0.8]$	[34]
HCLPIO	2018	$R = 0.2, ap = 0, bp = 0.25$	[70]
CPIO-C	2018	$R = 0.3, a = b = 0.5, c = 1.2495,$ $d = 1.0500, e = 2, f = -0.7, n = 0.95$	[27]
LFPIO	2018	$R = 0.2, rou = 0.0001, r = [0 \sim 1]$	[18]
SADE	2009	$Cr = 0.3, Strategy = 4, K = 4, LP = 50$	[51]
LSHADE	2014	$arc\_rate = 2.6, min\_popsize = 4$ $H = 5$	[60]
HCLPSO	2015	$a = 0, b = 0.25, c1 = [2.5 \sim 0.5],$ $c2 = [0.5 \sim 2.5], W = [0.99 \sim 0.2]$, $K = [3 \sim 1.5], g_1 = g_2 = 15$	[46]
DEBSO	2015	$p_{r0} = 0.2, p_{r1} = 0.8, p_{r11} = 0.4,$ $p_{r21} = 0.5, M = 5, CR = 0.5, F = 0.5$	[81]
MPEDE	2016	$Ng = 20, CR_m = 0.5, F_m = 0.5, l = 0.2$	[26]
PSO-DLS	2017	$c_1 = c_2 = 1.49445, N = 3,$ $M = 10, W = [0.9 \sim 0.4]$	[71]
DELMFO	2018	$CR = 0.7, R = rand(),$ $\sigma = 0.6, \delta = 0.15$	[41]
HCLDMS-PSO	2021	$W = [0.99 \sim 0.2], c_1 = [2.5 \sim 0.5],$ c2=[0.5∼2.5]	[65]
EESHHO	2021	All parameters adaptive	[42]
CMSRSSSA	2021	$c_1 = c_2$=[0,1], F_1=1, CR_1=0.1, F_2=0.8, CR_2=0.12,F_3=1, CR_3=0.9	[76]
HTNPIO		$CR_1 = 0.5(1 + rand), F_1 = rand,$ $CR_2 = 0.9, F_2 = 1, R = 0.25rand(1, D)$ $\mu = 0, \sigma_\theta = 1, \eta = 0.1$	

above, and the maximum number of the FEs is regarded as the termination criterion. Each of the six PIO methods repeatedly run 30 times on each CEC2017 function. The mean and the std value of errors obtained by each algorithm for 30 runs on each CEC2017 function are reported. Rank of each algorithm on each function is also calculated and displayed in the tables. Moreover, the Wilcoxon test is performed and shown following the rank value, a "(+)" represents HTNPIO is significantly better than the specific algorithm on the current problem, a "(-)" means HTNPIO is significantly worse than the specific algorithm on the current problem, while a "(≈)" suggests there are no significant difference between two algorithms.

Table 3 shows that HTNPIO provides the first final rank on 30 CEC2017 instances with $30 - D$ problems. The newly proposed HCLPIO wins the second place and LFPIO obtains the third rank. For 30 instances with $30 - D$, HTNPIO ranks first among PIOs for 26 times out of 29 functions, exhibiting extremely competitive performance, while HCLPIO provides the best solutions on the multimodel function F7, F8, and F10. However, the Wilcoxon test suggests HCLPIO only outperforms HTNPIO on F7, and F10, and shows similar performance with HTNPIO on F8. Additionally, HTNPIO also ranks behind CPIO-C and LFPIO on F10. Except for the 3 function, the Wilcoxon test showed that HTNPIO outperformed other PIO algorithms on other functions with a great advantage.

Table 4 shows that HTNPIO can provide the first final rank on 30 CEC2017 instances with $50 - D$ problems. Similarly to instances with 30-D, the HCLPIO wins the second place and LFPIO obtains the third rank, respectively. Among 30 CEC2017 instances with 50-D, HTNPIO ranked first for 25 times, while HCLPIO provides the best solutions on the multimodel function F5, F8 and F10, and LFPIO acquires the first rank on the unimodel F3. However, the wilcoxon test demonstrates that though LFPIO ranks before HTNPIO, there are no significant difference between two set of error data provided by LFPIO HTNPIO. comprehensively, the Wilcoxon test shows that HTNPIO outperformed other PIO algorithms with a great advantage on $50 - D$ problems.

Overall, HTNPIO beats other PIO algorithms by a huge margin in both 30 and 50 dimension problems in CEC2017

Table 3 Comparisons of HTNPIO and five PIO algorithms on 30 CEC2017 instances with dimension 30

Instances	Indicators	HCLPIO	LFPIO	CPIO-C	PIO	BQPIO	HTNPIO
F1	Mean	2.45E+03	2.79E+03	3.12E+07	1.33E+11	6.33E+11	**7.76E-05**
	Std	2.41E+03	3.91E+03	1.13E+08	1.79E+10	2.59E+09	**2.87E-04**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F3	Mean	2.55E+04	7.61E-01	2.98E+03	2.88E+10	2.89E+13	**3.27E-10**
	Std	1.07E+04	8.96E-01	6.19E+03	8.28E+10	2.19E+04	**8.90E-10**
	Rank	4(+)	2(+)	3(+)	5(+)	6(+)	**1**
F4	Mean	9.39E+01	8.22E+01	2.22E+02	5.40E+04	3.22E+05	**2.00E+01**
	Std	2.82E+01	2.59E+01	9.87E+01	1.56E+04	6.58E+02	**2.82E+01**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F5	Mean	6.73E+01	1.23E+02	1.23E+02	7.05E+02	2.65E+03	**3.94E+01**
	Std	1.01E+01	3.42E+01	3.41E+01	5.84E+01	3.11E+01	**1.23E+01**
	Rank	2(+)	3(+)	3(+)	5(+)	6(+)	**1**
F6	Mean	1.30E+00	2.38E-03	1.99E+01	1.43E+02	1.67E+02	**1.74E-13**
	Std	9.32E-01	1.01E-02	7.93E+00	1.28E+01	7.77E+00	**5.77E-14**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F7	Mean	**9.04E+01**	1.92E+02	2.18E+02	2.80E+03	1.30E+04	1.19E+02
	Std	**1.10E+01**	3.90E+01	5.35E+01	3.01E+02	7.66E+01	5.12E+01
	Rank	**1(-)**	3(+)	4(+)	5(+)	6(+)	2
F8	Mean	**6.70E+01**	1.16E+02	1.21E+02	6.40E+02	2.83E+03	7.47E+01
	Std	**1.33E+01**	2.67E+01	3.11E+01	5.14E+01	2.50E+01	2.76E+01
	Rank	**1(≈)**	3(+)	4(+)	5(+)	6(+)	2
F9	Mean	4.63E+02	2.79E+03	2.52E+03	4.05E+04	2.34E+05	**5.57E-01**
	Std	3.49E+02	1.82E+03	1.57E+03	6.19E+03	1.11E+03	**1.47E+00**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F10	Mean	**2.36E+03**	3.12E+03	3.73E+03	1.01E+04	3.60E+04	6.19E+03
	Std	**3.32E+02**	4.97E+02	5.95E+02	6.17E+02	2.74E+02	3.10E+02
	Rank	**1(-)**	2(-)	3(-)	5(+)	6(+)	4
F11	Mean	1.11E+02	1.25E+02	2.52E+02	3.14E+05	4.47E+09	**1.67E+01**
	Std	3.56E+01	4.71E+01	1.27E+02	8.56E+05	2.38E+02	**1.85E+01**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F12	Mean	1.11E+06	2.79E+05	9.92E+06	2.96E+10	3.83E+11	**1.59E+04**
	Std	7.57E+05	2.00E+05	1.89E+07	7.31E+09	2.91E+08	**1.21E+04**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F13	Mean	1.20E+04	1.29E+04	3.60E+04	3.14E+10	9.50E+10	**8.91E+01**
	Std	1.06E+04	1.27E+04	2.83E+04	1.12E+10	2.25E+07	**4.67E+01**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F14	Mean	1.57E+05	2.16E+04	2.19E+04	1.44E+08	1.06E+09	**3.75E+01**
	Std	2.19E+05	1.42E+04	6.87E+04	1.34E+08	1.08E+05	**1.20E+01**
	Rank	4(+)	2(+)	3(+)	5(+)	6(+)	**1**
F15	Mean	8.69E+02	7.37E+03	2.63E+04	9.17E+09	5.38E+10	**1.48E+01**
	Std	1.35E+03	8.00E+03	1.97E+04	3.46E+09	2.42E+06	**1.80E+01**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F16	Mean	8.94E+02	1.18E+03	1.01E+03	7.57E+03	4.48E+04	**6.87E+02**
	Std	1.60E+02	2.69E+02	2.59E+02	2.14E+03	3.00E+02	**5.07E+02**
	Rank	2(≈)	4(+)	3(+)	5(+)	6(+)	**1**
F17	Mean	2.50E+02	5.29E+02	3.59E+02	6.59E+04	1.67E+08	**7.95E+01**
	Std	1.40E+02	2.64E+02	1.49E+02	1.23E+05	1.84E+02	**6.44E+01**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**

Table 3 (continued)

Instances	Indicators	HCLPIO	LFPIO	CPIO-C	PIO	BQPIO	HTNPIO
F18	Mean	2.62E+05	1.63E+05	6.00E+05	1.11E+09	2.59E+09	**1.31E+02**
	Std	2.95E+05	1.12E+05	1.62E+06	6.44E+08	2.02E+06	**8.60E+01**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F19	Mean	2.30E+03	7.36E+03	1.04E+04	9.06E+09	5.68E+10	**1.60E+01**
	Std	3.06E+03	1.05E+04	1.19E+04	3.37E+09	7.97E+06	**5.66E+00**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F20	Mean	3.36E+02	5.37E+02	4.60E+02	2.00E+03	8.05E+03	**9.12E+01**
	Std	1.04E+02	2.10E+02	1.73E+02	1.78E+02	1.51E+02	**7.67E+01**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F21	Mean	2.71E+02	3.30E+02	3.20E+02	8.82E+02	3.50E+03	**2.34E+02**
	Std	1.23E+01	2.81E+01	2.18E+01	6.41E+01	3.75E+01	**2.12E+01**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F22	Mean	5.68E+02	2.35E+03	2.20E+03	1.03E+04	3.75E+04	**1.00E+02**
	Std	1.07E+03	1.91E+03	1.91E+03	5.16E+02	2.98E+02	**1.45E-13**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F23	Mean	4.44E+02	5.02E+02	4.62E+02	1.73E+03	6.35E+03	**3.82E+02**
	Std	2.33E+01	3.76E+01	4.24E+01	2.52E+02	4.05E+01	**2.49E+01**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F24	Mean	5.63E+02	6.27E+02	5.36E+02	2.05E+03	1.29E+04	**4.47E+02**
	Std	2.84E+01	7.06E+01	5.34E+01	3.72E+02	5.10E+01	**9.56E+00**
	Rank	3(+)	4(+)	2(+)	5(+)	6(+)	**1**
F25	Mean	4.07E+02	4.06E+02	4.76E+02	1.84E+04	1.46E+05	**3.86E+02**
	Std	1.43E+01	2.30E+01	5.07E+01	4.81E+03	1.28E+02	**2.01E+00**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F26	Mean	1.52E+03	2.19E+03	2.23E+03	1.42E+04	9.41E+04	**4.48E+02**
	Std	9.78E+02	1.36E+03	8.40E+02	2.52E+03	7.46E+02	**4.76E+02**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F27	Mean	5.56E+02	5.57E+02	5.83E+02	3.25E+03	1.52E+04	**5.03E+02**
	Std	1.60E+01	2.54E+01	2.89E+01	7.70E+02	7.36E+01	**6.78E+00**
	Rank	2(+)	3(+)	4(+)	5(+)	6(+)	**1**
F28	Mean	4.41E+02	4.02E+02	5.20E+02	1.21E+04	7.47E+04	**3.11E+02**
	Std	2.23E+01	3.69E+01	6.57E+01	2.79E+03	2.90E+02	**3.37E+01**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
F29	Mean	7.73E+02	9.46E+02	9.39E+02	1.31E+05	2.56E+07	**4.64E+02**
	Std	1.33E+02	2.56E+02	2.19E+02	2.34E+05	3.07E+02	**6.52E+01**
	Rank	2(+)	4(+)	3(+)	5(+)	6(+)	**1**
F30	Mean	6.72E+03	5.96E+03	7.91E+05	4.49E+09	8.04E+10	**3.01E+03**
	Std	2.31E+03	2.87E+03	8.40E+05	2.06E+09	8.54E+06	**6.25E+02**
	Rank	3(+)	2(+)	4(+)	5(+)	6(+)	**1**
Number of first rank		3	0	0	0	0	**26**
+/-/≈		25/2/2	28/1/0	28/1/0	29/0/0	29/0/0	
Final rank		2	3	4	5	6	**1**

test suite. The success of HTNPIO is mainly attributed to the effective interactions between different individuals. Further, the proper combination of PIO and differential evolutionary strategies not only enhances the global search ability of PIO, but also maintains the local exploitation capability. Therefore, the performance is greatly enhanced. Next, HTNPIO will be compared with other state-of-the-art swarm intelligence algorithms.

Table 4 Comparisons of HTNPIO and five PIO algorithms on 30 CEC2017 instances with dimension 50

Instances	Indicators	HCLPIO	LFPIO	CPIO-C	PIO	BQPIO	HTNPIO
F1	Mean	1.35E+05	4.05E+03	2.58E+08	2.59E+11	3.17E+10	**6.16E+02**
	Std	2.85E+03	4.72E+03	8.94E+08	2.66E+10	5.47E+09	**8.87E+02**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F3	Mean	3.30E+04	**2.79E+01**	4.25E+04	2.61E+11	1.03E+05	3.97E+01
	Std	5.92E+01	**2.40E+01**	3.55E+04	1.22E+12	2.92E+04	5.89E+01
	Rank	3(+)	**1(≈)**	4(+)	6(+)	5(+)	2
F4	Mean	1.32E+02	1.14E+02	6.81E+02	1.28E+05	7.36E+03	**5.38E+01**
	Std	4.71E+01	4.90E+01	2.62E+02	2.25E+04	2.00E+03	**3.31E+01**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F5	Mean	**1.42E+02**	2.80E+02	2.44E+02	1.24E+03	5.35E+02	2.48E+02
	Std	**2.69E+01**	5.45E+01	5.45E+01	7.30E+01	4.53E+01	6.97E+01
	Rank	**1(-)**	4(≈)	2(≈)	6(+)	5(+)	3
F6	Mean	1.18E+00	1.04E-02	3.17E+01	1.57E+02	6.71E+01	**7.77E-03**
	Std	1.08E-01	2.09E-02	9.61E+00	1.24E+01	5.52E+00	**4.25E-02**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F7	Mean	1.86E+02	4.26E+02	5.76E+02	5.45E+03	1.10E+03	**1.27E+02**
	Std	2.07E+01	8.13E+01	1.25E+02	4.43E+02	8.66E+01	**4.75E+01**
	Rank	2(+)	3(+)	4(+)	6(+)	5(+)	**1**
F8	Mean	**1.48E+02**	2.96E+02	2.56E+02	1.24E+03	5.38E+02	1.81E+02
	Std	**2.75E+01**	4.97E+01	5.88E+01	9.03E+01	2.95E+01	1.02E+02
	Rank	**1(≈)**	4(+)	3(+)	6(+)	5(+)	2
F9	Mean	3.71E+03	1.07E+04	9.65E+03	1.06E+05	1.98E+04	**6.27E+01**
	Std	4.44E+02	4.08E+03	2.94E+03	1.10E+04	4.08E+03	**1.56E+02**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**
F10	Mean	**4.60E+03**	5.71E+03	6.45E+03	1.73E+04	1.36E+04	1.01E+04
	Std	**8.81E+02**	7.62E+02	1.16E+03	6.76E+02	2.66E+02	2.94E+03
	Rank	**1(-)**	2(-)	3(-)	6(+)	5(+)	4
F11	Mean	2.93E+02	2.36E+02	5.70E+02	1.76E+06	3.30E+03	**5.11E+01**
	Std	2.40E+01	1.01E+02	3.01E+02	3.73E+06	1.11E+03	**1.65E+01**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F12	Mean	2.74E+06	1.93E+06	8.68E+07	1.36E+11	5.88E+09	**4.98E+04**
	Std	5.79E+04	1.02E+06	1.20E+08	3.28E+10	2.23E+09	**3.30E+04**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F13	Mean	1.63E+03	5.62E+03	5.01E+05	9.91E+10	4.97E+08	**1.31E+03**
	Std	3.63E+03	5.61E+03	2.32E+06	2.27E+10	2.98E+08	**1.87E+03**
	Rank	2(≈)	3(+)	4(+)	6(+)	5(+)	**1**
F14	Mean	9.29E+05	4.93E+04	1.31E+05	5.95E+08	1.81E+06	**7.34E+01**
	Std	6.81E+03	3.22E+04	1.71E+05	3.18E+08	1.54E+06	**2.78E+01**
	Rank	4(+)	2(+)	3(+)	6(+)	5(+)	**1**
F15	Mean	6.75E+03	7.37E+03	5.48E+04	3.48E+10	8.53E+07	**7.18E+01**
	Std	6.63E+03	6.52E+03	1.74E+05	8.84E+09	9.95E+07	**2.27E+01**
	Rank	2(+)	3(+)	4(+)	6(+)	5(+)	**1**
F16	Mean	1.49E+03	1.79E+03	1.79E+03	1.52E+04	3.44E+03	**7.94E+02**
	Std	3.69E+02	3.98E+02	3.66E+02	3.56E+03	6.70E+02	**2.57E+02**
	Rank	2(+)	3(+)	3(+)	6(+)	5(+)	**1**
F17	Mean	1.00E+03	1.53E+03	1.45E+03	4.22E+06	2.19E+03	**5.49E+02**
	Std	2.29E+02	3.28E+02	4.88E+02	4.63E+06	3.10E+02	**2.58E+02**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**

Table 4 (continued)

Instances	Indicators	HCLPIO	LFPIO	CPIO-C	PIO	BQPIO	HTNPIO
F18	Mean	1.34E+06	3.05E+05	3.20E+06	1.78E+09	1.20E+07	**9.73E+03**
	Std	3.13E+04	1.54E+05	3.88E+06	9.24E+08	7.57E+06	**1.29E+04**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F19	Mean	1.57E+04	1.67E+04	1.79E+04	1.47E+10	1.64E+07	**2.88E+01**
	Std	1.02E+04	1.32E+04	1.47E+04	4.42E+09	2.45E+07	**1.10E+01**
	Rank	2(+)	3(+)	4(+)	6(+)	5(+)	**1**
F20	Mean	7.83E+02	1.24E+03	1.11E+03	3.66E+03	1.89E+03	**3.13E+02**
	Std	2.35E+02	3.48E+02	3.07E+02	2.99E+02	2.33E+02	**1.40E+02**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**
F21	Mean	3.58E+02	4.99E+02	4.38E+02	1.51E+03	7.04E+02	**2.75E+02**
	Std	2.12E+01	5.85E+01	4.56E+01	1.20E+02	6.02E+01	**2.03E+01**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**
F22	Mean	5.35E+03	6.65E+03	7.00E+03	1.77E+04	1.36E+04	**5.08E+02**
	Std	2.67E+03	6.61E+02	1.11E+03	7.43E+02	7.47E+02	**2.24E+03**
	Rank	2(+)	3(+)	4(+)	6(+)	5(+)	**1**
F23	Mean	6.41E+02	7.98E+02	6.84E+02	3.13E+03	1.21E+03	**4.93E+02**
	Std	3.46E+01	9.09E+01	5.71E+01	4.67E+02	7.64E+01	**1.71E+01**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**
F24	Mean	8.76E+02	9.36E+02	7.78E+02	3.63E+03	1.36E+03	**5.64E+02**
	Std	2.71E+01	9.36E+01	7.49E+01	4.47E+02	8.24E+01	**1.99E+01**
	Rank	3(+)	4(+)	2(+)	6(+)	5(+)	**1**
F25	Mean	5.97E+02	5.45E+02	9.37E+02	6.32E+04	4.67E+03	**5.45E+02**
	Std	4.40E+01	3.12E+01	2.74E+02	1.08E+04	9.40E+02	**4.22E+01**
	Rank	3(+)	2(≈)	4(+)	6(+)	5(+)	**1**
F26	Mean	3.06E+03	4.44E+03	3.78E+03	2.99E+04	1.03E+04	**3.00E+02**
	Std	5.63E+02	2.44E+03	1.12E+03	4.30E+03	9.36E+02	**3.04E-13**
	Rank	2(+)	4(+)	3(+)	6(+)	5(+)	**1**
F27	Mean	8.66E+02	8.44E+02	9.53E+02	6.48E+03	2.22E+03	**5.63E+02**
	Std	4.16E+01	1.03E+02	9.81E+01	1.10E+03	2.34E+02	**3.11E+01**
	Rank	3(+)	2(+)	4(+)	6(+)	5(+)	**1**
F28	Mean	5.62E+02	4.98E+02	9.90E+02	2.30E+04	4.22E+03	**4.94E+02**
	Std	2.81E+01	3.34E+01	3.23E+02	3.31E+03	6.12E+02	**2.86E+01**
	Rank	3(+)	2(≈)	4(+)	6(+)	5(+)	**1**
F29	Mean	1.43E+03	1.43E+03	1.87E+03	3.03E+06	4.47E+03	**5.45E+02**
	Std	2.17E+02	3.44E+02	3.74E+02	3.35E+06	7.01E+02	**2.09E+02**
	Rank	2(+)	2(+)	4(+)	6(+)	5(+)	**1**
F30	Mean	9.89E+05	1.01E+06	5.65E+07	2.32E+10	3.06E+08	**6.10E+05**
	Std	2.31E+04	3.25E+05	3.57E+07	7.65E+09	1.14E+08	**2.41E+04**
	Rank	2(+)	3(+)	4(+)	6(+)	5(+)	**1**
Number of first rank		3	1	0	0	0	**25**
+/-/≈		25/2/2	24/1/4	27/1/1	29/0/0	29/0/0	/
Final rank		2	3	4	6	5	**1**

6.4 Comparison with other ten state-of-the-art algorithms

In this part, the proposed HTNPIO is further made comparisons with other ten algorithms (LSHADE, CMSRSSSA, EESHHO, SADE, MPEDE, DELMFO, HCLDMSPSO, HCLPSO, PSODLS and DEBSO) on the 30 CEC2017 problems with $30-D$, $50-D$. Note that for the sake of space, HCLDMSPSO is abbreviated as HDMSPSO. The parameters of all algorithms are listed in Table 2. The population

Table 5 Comparisons of HTNPIO and other ten advanced algorithms on 30 CEC2017 instances with dimension 30

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F1	Mean	7.76E-05	3.04E+03	9.67E-11	5.73E-10	2.52E+03	1.40E+02	7.48E-08	3.04E+03	3.43E+03	1.52E+03	**1.10E-13**
	Std	2.87E-04	4.21E+03	1.92E-10	3.13E-09	2.65E+03	2.84E+02	4.10E-07	4.16E+03	4.16E+03	2.03E+03	**3.03E-13**
	Rank	5	10 (+)	2 (-)	3 (-)	8 (+)	6 (+)	4 (-)	9 (+)	11 (+)	7 (+)	**1 (-)**
F3	Mean	3.27E-10	5.72E+02	**3.22E-14**	4.68E-13	1.70E-01	7.17E-02	6.42E-06	1.82E+03	1.70E+04	9.34E+00	6.37E-13
	Std	8.90E-10	5.66E+02	**4.14E-14**	3.22E-13	4.82E-01	9.57E-02	3.07E-05	1.65E+03	9.26E+03	9.45E+00	6.37E-13
	Rank	4	9 (+)	**1 (-)**	2 (-)	7 (+)	6 (+)	5 (≈)	10 (+)	11 (+)	8 (+)	3 (-)
F4	Mean	2.00E+01	6.98E+01	1.38E+01	**2.26E+00**	3.67E+01	5.05E+01	4.04E+01	8.32E+01	9.04E+01	7.57E+01	3.72E+01
	Std	2.82E+01	2.09E+01	2.43E+01	**2.01E+00**	3.28E+01	3.11E+01	3.59E+01	2.25E+01	2.61E+01	2.21E+01	2.97E+01
	Rank	3	8 (+)	2 (≈)	**1 (-)**	4 (+)	7 (+)	6 (+)	10 (+)	11 (+)	9 (+)	5 (≈)
F5	Mean	3.94E+01	3.09E+01	4.28E+01	7.05E+01	**2.68E+01**	3.94E+01	3.74E+01	1.90E+02	1.54E+02	2.80E+01	3.02E+01
	Std	1.23E+01	1.00E+01	1.09E+01	2.13E+01	**6.33E+00**	8.11E+00	8.93E+00	3.70E+01	3.51E+01	5.64E+00	6.65E+00
	Rank	6	4 (-)	8 (≈)	9 (+)	**1 (-)**	7 (≈)	5 (≈)	11 (+)	10 (+)	2 (-)	3 (-)
F6	Mean	**1.74E-13**	5.29E-07	3.33E-05	1.10E+00	6.83E-03	3.98E-13	3.64E-08	1.56E+01	3.56E+01	1.18E-03	5.05E-03
	Std	**5.77E-14**	2.62E-06	9.81E-05	1.40E+00	2.47E-02	1.02E-13	1.99E-07	9.62E+00	1.17E+01	1.14E-03	1.42E-02
	Rank	**1**	4 (≈)	5 (+)	9 (+)	8 (+)	2 (+)	3 (+)	10 (+)	11 (+)	6 (+)	7 (+)
F7	Mean	1.19E+02	6.02E+01	7.55E+01	1.23E+02	**5.26E+01**	8.15E+01	7.12E+01	4.08E+02	3.55E+02	5.64E+01	5.94E+01
	Std	5.12E+01	9.23E+00	9.50E+00	3.09E+01	**5.49E+00**	9.75E+00	6.84E+00	8.56E+01	9.18E+01	7.14E+00	7.23E+00
	Rank	8	4 (-)	6 (-)	9 (≈)	**1 (-)**	7 (-)	5 (-)	11 (+)	10 (+)	2 (-)	3 (-)
F8	Mean	7.47E+01	3.53E+01	4.16E+01	6.29E+01	**2.57E+01**	4.33E+01	3.79E+01	1.43E+02	1.38E+02	2.94E+01	3.04E+01
	Std	2.76E+01	8.81E+00	1.01E+01	1.52E+01	**5.92E+00**	9.86E+00	1.06E+01	2.34E+01	3.93E+01	6.61E+00	4.69E+00
	Rank	9	4 (-)	6 (-)	8 (≈)	**1 (-)**	7 (-)	5 (-)	11 (+)	10 (+)	2 (-)	3 (-)
F9	Mean	5.57E-01	1.11E+00	7.89E+00	2.27E+02	**7.45E-01**	1.73E+01	6.04E-02	3.60E+03	2.30E+03	**8.47E-12**	2.37E+00
	Std	1.47E+00	4.89E+00	1.43E+01	1.87E+02	**6.91E-01**	1.93E+01	1.38E-01	7.09E+02	7.59E+02	**3.44E-11**	2.48E+00
	Rank	3	5 (≈)	7 (+)	9 (+)	4 (+)	8 (+)	2 (-)	11 (+)	10 (+)	**1 (-)**	6 (+)
F10	Mean	6.19E+03	2.70E+03	2.27E+03	2.94E+03	2.20E+03	**2.01E+03**	2.42E+03	3.13E+03	3.70E+03	2.39E+03	2.09E+03
	Std	3.10E+02	5.89E+02	4.17E+02	5.44E+02	5.43E+02	**3.03E+02**	6.02E+02	5.10E+02	6.68E+02	3.85E+02	3.03E+02
	Rank	11	7 (-)	4 (-)	8 (-)	3 (-)	**1 (-)**	6 (-)	9 (-)	10 (-)	5 (-)	2 (-)
F11	Mean	**1.67E+01**	**2.32E+01**	7.56E+01	1.18E+02	5.20E+01	5.26E+01	2.50E+01	9.22E+01	1.89E+02	3.66E+01	8.79E+01
	Std	**1.85E+01**	**2.35E+01**	2.84E+01	5.04E+01	2.74E+01	2.32E+01	2.47E+01	3.38E+01	6.29E+01	1.72E+01	4.77E+01
	Rank	**1**	2 (≈)	7 (+)	10 (+)	5 (+)	6 (+)	3 (≈)	9 (+)	11 (+)	4 (+)	8 (+)
F12	Mean	1.59E+04	2.11E+04	**7.74E+03**	8.10E+03	1.76E+04	1.01E+05	1.22E+04	1.15E+06	1.06E+06	8.90E+04	**4.79E+03**
	Std	1.21E+04	1.18E+04	**3.62E+03**	9.11E+03	1.13E+04	1.17E+05	7.69E+03	1.03E+06	8.33E+05	8.10E+04	**4.08E+03**
	Rank	5	7 (≈)	2 (-)	3 (-)	6 (≈)	9 (+)	4 (≈)	11 (+)	10 (+)	8 (+)	**1 (-)**

Table 5 (continued)

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F13	Mean	**8.91E+01**	9.27E+03	1.07E+04	7.44E+03	8.29E+03	3.74E+02	6.33E+02	1.09E+04	9.83E+04	2.89E+03	4.31E+02
	Std	**4.67E+01**	8.42E+03	6.88E+03	1.46E+04	6.27E+03	3.23E+02	8.16E+02	1.21E+04	7.56E+04	3.10E+03	4.76E+02
	Rank	**1**	8 (+)	9 (+)	6 (+)	7 (+)	2 (+)	4 (+)	10 (+)	11 (+)	5 (+)	3 (+)
F14	Mean	**3.75E+01**	**3.81E+01**	7.29E+01	2.01E+02	3.01E+03	6.05E+03	6.90E+01	3.83E+04	1.13E+04	3.23E+03	1.41E+02
	Std	**1.20E+01**	**2.24E+01**	3.55E+01	9.08E+01	2.68E+03	6.80E+03	3.11E+01	3.16E+04	7.97E+03	3.32E+03	5.75E+01
	Rank	**1**	2 (≈)	4 (+)	6 (+)	7 (+)	9 (+)	3 (+)	11 (+)	10 (+)	8 (+)	5 (+)
F15	Mean	**1.48E+01**	4.07E+02	1.68E+02	3.13E+02	1.78E+03	2.42E+02	3.67E+01	3.46E+03	5.32E+04	2.73E+03	2.23E+02
	Std	**1.80E+01**	1.36E+03	1.10E+02	2.51E+02	2.15E+03	4.83E+02	4.56E+01	3.99E+03	4.16E+04	4.28E+03	1.40E+02
	Rank	**1**	7 (+)	3 (+)	6 (+)	8 (+)	5 (+)	2 (+)	10 (+)	11 (+)	9 (+)	4 (+)
F16	Mean	6.87E+02	4.31E+02	4.48E+02	7.55E+02	5.07E+02	5.80E+02	**2.04E+02**	1.11E+03	1.30E+03	3.21E+02	5.25E+02
	Std	5.07E+02	1.59E+02	1.81E+02	2.90E+02	1.75E+02	1.21E+02	**1.74E+02**	2.51E+02	3.79E+02	1.62E+02	1.44E+02
	Rank	8	3 (-)	4 (-)	9 (≈)	5 (≈)	7 (≈)	**1 (-)**	10 (+)	11 (+)	2 (-)	6 (-)
F17	Mean	7.95E+01	6.53E+01	**5.90E+01**	3.09E+02	1.61E+02	1.09E+02	9.56E+01	5.18E+02	5.95E+02	6.81E+01	1.23E+02
	Std	6.44E+01	5.96E+01	**5.40E+01**	1.82E+02	8.13E+01	5.46E+01	4.80E+01	2.10E+02	2.64E+02	3.20E+01	7.48E+01
	Rank	4	2 (≈)	**1 (≈)**	9 (+)	8 (+)	6 (+)	5 (≈)	10 (+)	11 (+)	3 (≈)	7 (+)
F18	Mean	**1.31E+02**	3.30E+04	6.32E+03	**1.77E+03**	1.13E+05	1.00E+05	1.89E+04	3.03E+05	2.27E+05	1.06E+05	4.62E+02
	Std	**8.60E+01**	2.10E+04	5.92E+03	**1.44E+03**	6.37E+04	7.32E+04	1.89E+04	2.65E+05	1.12E+05	8.80E+04	6.41E+02
	Rank	**1**	6 (+)	4 (+)	3 (+)	9 (+)	7 (+)	5 (+)	11 (+)	10 (+)	8 (+)	2 (+)
F19	Mean	**1.60E+01**	5.75E+02	**5.13E+01**	1.88E+02	4.55E+03	1.45E+02	2.23E+01	4.09E+03	7.81E+04	4.08E+03	1.31E+02
	Std	**5.66E+00**	2.88E+03	**2.70E+01**	2.93E+02	3.48E+03	2.31E+02	1.19E+01	4.18E+03	4.26E+04	7.95E+03	6.55E+01
	Rank	**1**	7 (≈)	3 (+)	6 (+)	10 (+)	5 (+)	2 (≈)	9 (+)	11 (+)	8 (+)	4 (+)
F20	Mean	**9.12E+01**	1.48E+02	1.11E+02	3.64E+02	2.16E+02	1.53E+02	9.85E+01	4.84E+02	7.27E+02	1.45E+02	1.83E+02
	Std	**7.67E+01**	1.22E+02	6.05E+01	1.61E+02	5.89E+01	8.94E+01	8.01E+01	1.79E+02	2.20E+02	7.27E+01	7.45E+01
	Rank	**1**	5 (≈)	3 (≈)	9 (+)	8 (+)	6 (+)	2 (≈)	10 (+)	11 (+)	4 (+)	7 (+)
F21	Mean	2.34E+02	2.31E+02	2.39E+02	2.63E+02	2.34E+02	**2.18E+02**	2.37E+02	3.55E+02	3.52E+02	2.28E+02	2.33E+02
	Std	2.12E+01	5.81E+00	1.22E+01	1.48E+01	8.33E+00	**6.04E+01**	8.36E+00	6.46E+01	4.01E+01	6.61E+00	5.32E+00
	Rank	5	3 (≈)	8 (+)	9 (+)	6 (≈)	**1 (≈)**	7 (≈)	11 (+)	10 (+)	2 (≈)	4 (≈)
F22	Mean	1.00E+02	1.62E+02	1.00E+02	1.60E+03	**1.00E+02**	1.00E+02	1.00E+02	1.39E+03	3.29E+03	1.00E+02	1.00E+02
	Std	1.45E-13	3.41E+02	7.49E-01	1.70E+03	**6.28E-01**	1.09E+00	2.85E-13	1.92E+03	1.85E+03	1.85E-10	7.15E-01
	Rank	4	8 (-)	1 (-)	10 (+)	**1 (≈)**	1 (+)	4 (+)	9 (+)	11 (+)	6 (+)	7 (+)

Table 5 (continued)

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F23	Mean	3.82E+02	3.83E+02	3.96E+02	4.46E+02	**3.77E+02**	3.99E+02	3.90E+02	5.45E+02	6.71E+02	3.82E+02	3.81E+02
	Std	2.49E+01	1.18E+01	1.50E+01	2.60E+01	**8.42E+00**	1.07E+01	1.31E+01	4.97E+01	8.22E+01	1.19E+01	5.90E+00
	Rank	4	5 (≈)	7 (+)	9 (+)	**1** (≈)	8 (+)	6 (+)	10 (+)	11 (+)	3 (≈)	2 (≈)
F24	Mean	4.47E+02	4.50E+02	4.67E+02	5.16E+02	**4.39E+02**	4.55E+02	4.55E+02	6.85E+02	7.68E+02	4.45E+02	4.51E+02
	Std	9.56E+00	7.38E+00	1.34E+01	2.00E+01	**7.76E+00**	8.53E+01	1.37E+01	8.72E+01	9.67E+01	8.16E+00	1.04E+01
	Rank	3	4 (+)	8 (+)	9 (+)	**1 (-)**	6 (+)	7 (+)	10 (+)	11 (+)	2 (≈)	5 (≈)
F25	Mean	3.86E+02	3.87E+02	3.91E+02	**3.81E+02**	3.91E+02	3.87E+02	3.88E+02	3.99E+02	3.87E+02	3.87E+02	3.87E+02
	Std	2.01E+00	7.68E-01	1.08E+01	**6.80E+00**	1.16E+01	1.19E+00	6.86E+00	1.35E+01	1.78E+00	5.69E-01	2.46E+00
	Rank	2	4 (+)	9 (+)	**1 (-)**	9 (+)	4 (+)	8 (+)	11 (+)	4 (≈)	3 (+)	7 (+)
F26	Mean	4.48E+02	1.38E+03	1.18E+03	1.91E+03	7.11E+02	**2.49E+02**	1.27E+03	3.31E+03	4.18E+03	1.21E+03	1.35E+03
	Std	4.76E+02	1.26E+02	6.13E+02	5.17E+02	4.65E+02	**4.93E+01**	1.15E+02	1.33E+03	1.28E+03	9.21E+01	2.68E+02
	Rank	2	8 (+)	4 (+)	9 (+)	3 (≈)	**1 (-)**	6 (+)	10 (+)	11 (+)	5 (+)	7 (+)
F27	Mean	5.03E+02	5.10E+02	5.19E+02	**5.00E+02**	5.19E+02	5.12E+02	5.00E+02	5.79E+02	7.82E+02	5.13E+02	5.10E+02
	Std	6.78E+00	9.06E+00	8.30E+00	**2.65E-04**	1.05E+01	5.96E+00	6.83E+00	3.91E+01	1.28E+02	7.48E+00	9.50E+00
	Rank	3	4 (+)	8 (+)	**1 (-)**	8 (+)	6 (+)	2 (-)	10 (+)	11 (+)	7 (+)	5 (+)
F28	Mean	**3.11E+02**	3.56E+02	**3.14E+02**	4.43E+02	3.26E+02	3.78E+02	3.38E+02	3.88E+02	3.65E+02	3.74E+02	3.37E+02
	Std	**3.37E+01**	6.27E+01	**3.60E+01**	7.26E+01	4.40E+01	3.90E+01	6.03E+01	4.45E+01	5.66E+01	4.31E+01	5.91E+01
	Rank	**1**	6 (+)	2 (-)	11 (+)	3 (+)	9 (+)	5 (+)	10 (+)	7 (+)	8 (+)	4 (+)
F29	Mean	4.64E+02	**4.53E+02**	4.73E+02	7.03E+02	5.64E+02	5.10E+02	4.67E+02	9.97E+02	1.38E+03	5.01E+02	5.35E+02
	Std	6.52E+01	**5.03E+01**	5.34E+01	1.34E+02	9.74E+01	5.42E+01	6.64E+01	2.55E+02	2.74E+02	3.45E+01	7.77E+01
	Rank	2	**1** (≈)	4 (≈)	9 (+)	8 (+)	6 (+)	3 (≈)	10 (+)	11 (+)	5 (+)	7 (+)
F30	Mean	3.01E+03	4.16E+03	3.65E+03	**3.36E+02**	4.50E+03	3.82E+03	2.67E+03	7.58E+03	3.35E+05	3.67E+03	2.27E+03
	Std	6.25E+02	1.25E+03	1.52E+03	**7.54E+01**	1.66E+03	1.10E+03	5.24E+02	4.31E+03	2.47E+05	9.45E+02	2.14E+02
	Rank	4	8 (+)	5 (+)	**1 (-)**	9 (+)	7 (+)	3 (-)	10 (+)	11 (+)	6 (+)	2 (-)
Number of first rank		**9**	3	5	5	7	3	1	0	0	1	2
+/-/≈		/	12/11/6	15/5/9	18/3/8	18/6/5	22/3/4	12/9/8	28/0/1	27/1/1	19/4/6	16/4/9
Average rank		**3.59**	5.34	4.72	6.69	5.48	5.59	4.24	10.14	10.31	5.10	4.48
Final rank		**1**	6	4	9	7	8	2	10	11	5	3

Table 6 Comparisons of HTNPIO and other ten advanced algorithms on 30 CEC2017 instances with dimension 50

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F1	Mean	6.16E+02	4.06E+03	1.44E+03	5.24E+02	3.24E+03	2.09E+02	6.88E+02	4.56E+03	4.40E+03	2.51E+03	**5.85E-01**
	Std	8.87E+02	4.87E+03	2.10E+03	1.74E+03	5.13E+03	4.38E+02	1.26E+03	6.67E+03	3.85E+03	2.77E+03	**1.56E+00**
	Rank	4	9(+)	6(+)	3(-)	8(+)	2(≈)	5(≈)	11(+)	10(+)	7(+)	**1(-)**
F3	Mean	3.97E+01	1.84E+04	1.01E-01	**5.16E-12**	8.24E+02	3.00E+02	1.03E+01	8.59E+03	5.75E+04	4.86E+03	7.74E-12
	Std	5.89E+01	8.53E+03	2.98E-01	**8.71E-12**	4.72E+02	2.42E+02	2.29E+01	3.34E+03	1.57E+04	1.77E+03	9.53E-12
	Rank	5	10(+)	3(-)	**1(-)**	7(+)	6(+)	4(-)	9(+)	11(+)	8(+)	2(-)
F4	Mean	5.38E+01	6.37E+01	6.82E+01	**9.94E+00**	9.27E+01	7.59E+01	7.43E+01	1.25E+02	1.41E+02	9.01E+01	4.35E+01
	Std	3.31E+01	4.45E+01	4.16E+01	**1.61E+01**	4.41E+01	3.67E+01	4.66E+01	4.95E+01	5.63E+01	4.76E+01	4.75E+01
	Rank	3	4(≈)	5(≈)	**1(-)**	9(+)	7(+)	6(≈)	10(+)	11(+)	8(+)	2(≈)
F5	Mean	2.48E+02	6.38E+01	1.03E+02	1.78E+02	**6.07E+01**	1.02E+02	8.08E+01	3.46E+02	2.85E+02	6.37E+01	7.67E+01
	Std	6.97E+01	1.32E+01	1.40E+01	2.87E+01	**1.18E+01**	1.76E+01	1.36E+01	3.74E+01	6.13E+01	9.79E+00	1.22E+01
	Rank	9	3(-)	7(-)	8(-)	**1(-)**	6(-)	5(-)	11(+)	10(≈)	2(-)	4(-)
F6	Mean	7.77E-03	1.26E-04	7.32E-03	8.26E+00	8.97E-03	**7.16E-13**	1.68E-08	2.09E+01	4.33E+01	9.18E-03	2.04E-01
	Std	4.25E-02	3.53E-04	1.29E-02	4.19E+00	7.44E-03	**1.56E-13**	5.73E-08	1.07E+01	9.58E+00	4.44E-03	2.01E-01
	Rank	5	3(+)	4(+)	9(+)	6(+)	**1(+)**	2(+)	10(+)	11(+)	7(+)	8(+)
F7	Mean	1.27E+02	1.21E+02	1.76E+02	3.27E+02	1.06E+02	1.70E+02	1.38E+02	7.95E+02	7.95E+02	**1.05E+02**	1.43E+02
	Std	4.75E+01	1.29E+01	2.31E+01	8.28E+01	1.22E+01	1.85E+01	1.32E+01	1.34E+02	1.37E+02	**1.06E+01**	1.82E+01
	Rank	4	3(≈)	8(+)	9(+)	2(-)	7(+)	5(+)	10(+)	11(+)	**1(-)**	6(+)
F8	Mean	1.81E+02	**6.36E+01**	1.09E+02	1.66E+02	6.77E+01	1.05E+02	7.43E+01	3.21E+02	2.85E+02	**6.10E+01**	7.68E+01
	Std	1.02E+02	**1.29E+01**	2.42E+01	3.11E+01	1.43E+01	1.74E+01	1.52E+01	3.79E+01	5.72E+01	**1.11E+01**	1.23E+01
	Rank	9	2(-)	7(-)	8(≈)	3(-)	6(-)	4(-)	11(+)	10(+)	**1(-)**	5(-)
F9	Mean	6.27E+01	5.43E+00	3.30E+02	2.94E+03	9.26E+00	3.50E+02	8.67E-01	1.08E+04	7.22E+03	**1.33E-01**	1.89E+02
	Std	1.56E+02	5.42E+00	2.40E+02	1.27E+03	4.89E+00	2.46E+02	1.09E+00	1.51E+03	1.79E+03	**2.19E-01**	1.95E+02
	Rank	5	3(-)	7(+)	9(+)	4(≈)	8(+)	2(-)	11(+)	10(+)	**1(-)**	6(+)
F10	Mean	1.01E+04	5.51E+03	4.27E+03	5.57E+03	4.49E+03	4.00E+03	4.98E+03	5.68E+03	6.58E+03	4.50E+03	**3.84E+03**
	Std	2.94E+03	9.26E+02	7.20E+02	9.01E+02	7.68E+02	4.36E+02	8.18E+02	7.17E+02	8.39E+02	4.05E+02	**4.50E+02**
	Rank	11	7(-)	3(-)	8(-)	4(-)	2(-)	6(-)	9(-)	10(-)	5(-)	**1(-)**
F11	Mean	5.11E+01	**4.48E+01**	1.16E+02	2.66E+02	9.57E+01	1.43E+02	5.02E+01	1.30E+02	2.71E+02	9.54E+01	2.15E+02
	Std	1.65E+01	**1.33E+01**	3.88E+01	9.85E+01	2.51E+01	3.65E+01	1.26E+01	2.43E+01	8.23E+01	1.97E+01	5.13E+01
	Rank	3	**1(≈)**	6(+)	10(+)	5(+)	8(+)	2(≈)	7(+)	11(+)	4(+)	9(+)
F12	Mean	4.98E+04	3.63E+05	4.90E+04	3.58E+04	2.11E+05	8.64E+05	9.02E+04	3.13E+06	5.35E+06	8.21E+05	**1.73E+04**
	Std	3.30E+04	2.06E+05	2.15E+04	3.03E+04	1.36E+05	4.86E+05	6.37E+04	1.80E+06	2.69E+06	4.45E+05	**1.29E+04**
	Rank	4	7(+)	3(≈)	2(≈)	6(+)	9(+)	5(+)	10(+)	11(+)	8(+)	**1(-)**

Table 6 (continued)

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F13	Mean	1.31E+03	3.55E+03	1.10E+03	1.88E+03	2.29E+03	**5.29E+02**	4.42E+03	9.23E+03	9.77E+04	2.66E+03	2.26E+03
	Std	1.87E+03	4.64E+03	1.24E+03	2.54E+03	3.02E+03	**3.19E+02**	6.07E+03	7.32E+03	4.60E+04	2.92E+03	2.17E+03
	Rank	3	8(+)	2(≈)	4(≈)	6(≈)	**1**(≈)	9(+)	10(+)	11(+)	7(≈)	5(≈)
F14	Mean	**7.34E+01**	7.81E+03	1.26E+03	2.12E+03	1.83E+04	3.17E+04	7.85E+02	9.44E+04	3.72E+04	3.02E+04	3.53E+02
	Std	**2.78E+01**	1.07E+04	9.78E+02	1.92E+03	1.27E+04	1.82E+04	6.11E+02	7.34E+04	2.44E+04	2.41E+04	1.12E+02
	Rank	**1**	6(+)	4(+)	5(+)	7(+)	9(+)	3(+)	11(+)	10(+)	8(+)	2(+)
F15	Mean	**7.18E+01**	2.88E+03	3.98E+03	3.23E+03	4.58E+03	3.98E+02	4.95E+02	1.41E+04	4.62E+04	2.63E+03	4.16E+02
	Std	**2.27E+01**	3.56E+03	3.14E+03	7.00E+03	4.02E+03	4.34E+02	1.14E+03	6.03E+03	2.36E+04	2.92E+03	1.82E+02
	Rank	**1**	6(+)	8(+)	7(+)	9(+)	2(+)	4(≈)	10(+)	11(+)	5(+)	3(+)
F16	Mean	7.94E+02	9.91E+02	9.79E+02	1.45E+03	6.61E+02	1.18E+03	**6.24E+02**	1.85E+03	2.21E+03	7.34E+02	9.81E+02
	Std	2.57E+02	2.23E+02	2.65E+02	3.98E+02	2.60E+02	2.09E+02	**2.24E+02**	4.71E+02	5.02E+02	2.02E+02	1.37E+02
	Rank	4	7(+)	5(+)	9(+)	2(-)	8(+)	**1(-)**	10(+)	11(+)	3(≈)	6(+)
F17	Mean	5.49E+02	6.47E+02	6.71E+02	1.17E+03	6.11E+02	8.71E+02	**4.71E+02**	1.57E+03	1.68E+03	5.63E+02	7.83E+02
	Std	2.58E+02	2.63E+02	2.04E+02	2.60E+02	1.52E+02	1.74E+02	**1.73E+02**	2.95E+02	3.50E+02	1.18E+02	1.65E+02
	Rank	2	5(≈)	6(+)	9(+)	4(≈)	8(+)	**1**(≈)	10(+)	11(+)	3(≈)	7(+)
F18	Mean	9.73E+03	1.71E+05	1.27E+04	1.39E+04	1.43E+05	1.68E+05	1.61E+04	5.28E+05	2.59E+05	3.04E+05	**2.10E+03**
	Std	1.29E+04	1.40E+05	8.71E+03	2.30E+04	1.21E+05	1.48E+05	1.16E+04	3.89E+05	1.70E+05	2.01E+05	**2.01E+03**
	Rank	2	8(+)	3(≈)	4(≈)	6(+)	7(+)	5(+)	11(+)	9(+)	10(+)	**1(-)**
F19	Mean	**2.88E+01**	1.11E+04	1.26E+04	4.02E+03	1.46E+04	5.66E+02	1.55E+02	1.87E+04	2.27E+05	1.19E+04	1.88E+02
	Std	**1.10E+01**	7.85E+03	5.49E+03	6.27E+03	4.56E+03	6.72E+02	2.70E+02	1.33E+04	8.09E+04	7.43E+03	6.78E+01
	Rank	**1**	6(+)	8(+)	5(+)	9(+)	4(+)	2(+)	10(+)	11(+)	7(+)	3(+)
F20	Mean	**3.13E+02**	5.04E+02	4.05E+02	8.79E+02	3.65E+02	5.86E+02	4.40E+02	1.21E+03	1.49E+03	3.94E+02	4.61E+02
	Std	**1.40E+02**	2.05E+02	1.98E+02	2.86E+02	1.08E+02	1.95E+02	2.42E+02	2.88E+02	3.75E+02	1.51E+02	1.57E+02
	Rank	**1**	7(+)	4(≈)	9(+)	2(≈)	8(+)	5(+)	10(+)	11(+)	3(+)	6(+)
F21	Mean	2.75E+02	2.66E+02	2.94E+02	3.56E+02	2.65E+02	3.11E+02	2.78E+02	5.28E+02	5.22E+02	**2.62E+02**	2.79E+02
	Std	2.03E+01	1.24E+01	1.58E+01	2.76E+01	1.43E+01	2.19E+01	2.05E+01	5.07E+01	6.51E+01	**1.30E+01**	2.12E+01
	Rank	4	3(≈)	7(+)	9(+)	2(-)	8(+)	5(≈)	11(+)	10(+)	**1(-)**	6(≈)

Table 6 (continued)

Instances	Indicators	HTNPIO	DELMFO	SADE	MPEDE	PSODLS	HCLPSO	CMSRSSSA	EESHHO	DEBSO	HDMSPSO	LSHADE
F22	Mean	**5.08E+02**	5.56E+03	4.39E+03	6.76E+03	2.37E+03	3.70E+03	3.80E+03	6.80E+03	7.11E+03	3.70E+03	4.28E+03
	Std	**2.24E+03**	1.64E+03	1.78E+03	1.12E+03	2.54E+03	1.86E+03	2.37E+03	9.71E+02	1.02E+03	2.26E+03	1.46E+03
	Rank	**1**	8(+)	7(+)	9(+)	2(+)	4(+)	5(+)	10(+)	11(+)	3(+)	6(+)
F23	Mean	4.93E+02	4.96E+02	5.40E+02	6.47E+02	4.89E+02	5.57E+02	**4.86E+02**	8.53E+02	1.14E+03	4.94E+02	5.22E+02
	Std	1.71E+01	1.67E+01	2.08E+01	5.78E+01	1.79E+01	2.73E+01	**2.16E+01**	7.35E+01	1.92E+02	2.03E+01	2.46E+01
	Rank	3	5(≈)	7(+)	9(+)	2(≈)	8(+)	**1(≈)**	10(+)	11(+)	4(≈)	6(+)
F24	Mean	5.64E+02	5.59E+02	6.19E+02	7.51E+02	**5.49E+02**	6.31E+02	5.59E+02	1.13E+03	1.33E+03	5.61E+02	5.87E+02
	Std	1.99E+01	1.55E+01	2.53E+01	4.66E+01	**1.98E+01**	3.19E+01	1.70E+01	1.35E+02	1.90E+02	1.70E+01	2.06E+01
	Rank	5	2(≈)	7(+)	9(+)	**1(-)**	8(+)	3(≈)	10(+)	11(+)	4(≈)	6(+)
F25	Mean	5.45E+02	5.21E+02	5.56E+02	**4.49E+02**	5.67E+02	5.08E+02	5.51E+02	5.56E+02	5.47E+02	5.12E+02	5.55E+02
	Std	4.22E+01	3.63E+01	4.22E+01	**2.20E+01**	2.50E+01	2.72E+01	3.45E+01	3.54E+01	4.19E+01	2.42E+01	3.33E+01
	Rank	5	4(≈)	9(≈)	**1(-)**	11(+)	2(-)	7(≈)	10(≈)	6(≈)	3(-)	8(≈)
F26	Mean	**3.00E+02**	1.91E+03	2.97E+03	3.64E+03	1.01E+03	1.20E+03	1.73E+03	5.53E+03	8.18E+03	1.74E+03	2.13E+03
	Std	**3.04E-13**	1.57E+02	3.33E+02	7.56E+02	6.48E+02	1.07E+03	1.76E+02	3.44E+03	2.00E+03	3.13E+02	2.22E+02
	Rank	**1**	6(+)	8(+)	9(+)	2(+)	3(+)	4(+)	10(+)	11(+)	5(+)	7(+)
F27	Mean	5.63E+02	5.87E+02	7.24E+02	**5.00E+02**	6.45E+02	6.31E+02	5.78E+02	8.97E+02	1.70E+03	6.04E+02	6.14E+02
	Std	3.11E+01	4.03E+01	5.49E+01	**2.73E-04**	4.19E+01	3.25E+01	4.32E+01	1.46E+02	3.36E+02	3.02E+01	4.47E+01
	Rank	2	4(+)	9(+)	**1(-)**	8(+)	7(+)	3(≈)	10(+)	11(+)	5(+)	6(+)
F28	Mean	4.94E+02	4.97E+02	5.05E+02	4.98E+02	4.97E+02	4.87E+02	5.03E+02	5.05E+02	5.04E+02	**4.73E+02**	4.96E+02
	Std	2.86E+01	1.68E+01	2.48E+01	7.38E+00	1.72E+01	2.06E+01	8.56E+00	3.54E+01	1.62E+01	**2.10E+01**	1.88E+01
	Rank	3	5(≈)	10(+)	7(≈)	5(≈)	2(≈)	8(+)	11(≈)	9(+)	**1(-)**	4(≈)
F29	Mean	5.45E+02	**5.17E+02**	6.36E+02	1.37E+03	6.95E+02	7.36E+02	**4.06E+02**	1.44E+03	2.52E+03	6.04E+02	7.52E+02
	Std	2.09E+02	**1.59E+02**	1.17E+02	4.49E+02	1.36E+02	1.79E+02	**5.77E+01**	4.11E+02	5.88E+02	1.25E+02	1.65E+02
	Rank	3	2(≈)	5(+)	9(+)	6(+)	7(+)	1(-)	10(+)	11(+)	4(≈)	8(+)
F30	Mean	6.10E+05	7.97E+05	8.50E+05	**7.83E+02**	8.75E+05	7.05E+05	6.09E+05	9.70E+05	1.10E+07	8.28E+05	7.07E+05
	Std	2.41E+04	1.04E+05	1.09E+05	**7.96E+02**	9.43E+04	6.24E+04	2.93E+04	2.15E+05	2.15E+06	1.04E+05	1.11E+05
	Rank	3	6(+)	8(+)	**1(-)**	9(+)	4(+)	2(≈)	10(+)	11(+)	7(+)	5(+)
Number of first rank		**6**	3	0	5	2	2	4	0	0	5	4
+/-/ydy		/	15/4/10	19/4/6	16/8/5	16/7/6	22/4/3	11/7/11	26/1/2	26/1/2	15/8/6	17/7/5
Average rank		**3.69**	5.04	6.07	6.46	5.00	5.71	3.93	10.07	10.46	4.57	4.96
Final rank		**1**	6	8	9	5	7	2	10	11	3	4

size and maximum number of FEs are fairly set as 30 and 10000D for all compared algorithms. All 11 algorithms are executed for 30 times on each CEC2017 instance. For comparison, the mean and std of errors obtained by each algorithm are reported. What's more, rank of each algorithm on each function is also shown in the tables. For each indicator of an instance in Tables 5, 6 and 7, the best result is bolded. Additionally, a "(+)" performed by the wilcoxon test represents that HTNPIO is significantly better than the specific algorithm on the current problem, a "(-)" means HTNPIO is significantly worse than the specific algorithm on the current problem, while a "(≈)" suggests there are no significant difference between two algorithms.

Comparison results of 11 algorithms on $30-D$ CEC2017 instances are shown in Table 5. Overall, the proposed HTNPIO beat all the compared algorithms including extremely competitive LSHADE. The average rank of HTNPIO is 3.59 and the final rank of HTNPIO is 1. The newly presented CMSRSSSA ranks second, and defeats LSHADE by a narrow margin. What's more, HTNPIO obtains 9 first ranks out of 29 functions, which is the most among all algorithms. It includes a multi-model problem F6, hybrid functions F11, F13, F14, F15, F18, F19 and F20, and a composition F28. It means that HTNPIO has a very prominent advantage in dealing with the complex problem. Moreover, on the unimodel problems F1 and F3, HTNPIO ranks 5th and 4th. It means that the local exploitation capability of HTNPIO is also not bad compared with the advanced algorithms. Specifically, from wilcoxon test, HTNPIO can provide better solutions than LSHADE on 16 functions and provide worse solutions than LSHADE on 4 functions. Further, HTNPIO and LSHADE obtain similar solutions on 9 functions.

Table 6 lists the comparative results of 11 algorithms on $50-D$ CEC2017 functions. On the $50-D$ problems, HTNPIO can also has the minimum average and final rank over all other algorithms. Newly presented CMSRSSSA and HCLDMSPSO rank 2nd and 3rd, respectively. LSHADE obtains the 4th rank in this experiment. What's more, HTNPIO gains 6 first ranks, including 4 hybrid functions F14-15, F19-20, and 2 compositions F22 and F26, followed by MPEDE and HCLDMSPSO both with 5 first ranks. Hence, it gives a demonstration of the good performance of HTNPIO on $50-D$ CEC2017 functions.

To sum up, HTNPIO has the best performance on both $30-D$ and $50-D$ CEC2017 problems compared with LSHADE, CMSRSSSA, EESHHO, SADE, MPEDE, DELMFO, HCLDMSPSO, HCLPSO, PSODLS and DEBSO. Precisely, HTNPIO has a prominent advantage on the hybrid and composition problems on both $30-D$ and $50-D$ CEC2017 problems, verifying that the proposed strategies successfully enhanced the exploration ability of PIO. What's more, the performance of HTNPIO on unimodel problems is also not bad, and the solutions of HTNPIO to most CEC2017 problems is accurate. It demonstrates that the proposed combination of SMS, LMS and ELS is effective, and the exploitation capability is also maintained. Thus, HTNPIO can provide extremely competitive performance.

6.5 Statistical distribution analysis

The reported mean and standard deviation value of the errors could not reveal the distribution of the optimal data for 30 runs in details. To make up for this defect, we utilized the violin plot to illustrate the distribution of the optimal results by different algorithms. Figures 8 and 9 respectively shows the error of the optimal results for all functions of $30-D$ and $50-D$ in the CEC2017 test suite obtained from the 11 algorithms over 30 independent runs. Note that in order to make the comparison results clearer and easier to display, all errors take the logarithm of 10 (log_{10}). What's more, F25 and F30 are not shown in figures because the comparison results provided by all algorithms cannot be distinctly differed in the violin plot.

Figure 8 shows the statistical distribution of 30-D error values of 30 runs obtained by 11 advanced algorithms. Overall, the proposed HTNPIO can provide consistent distribution for most functions. Specifically, HTNPIO shows extremely competitive performance on F3, F6, F9, F11, F13-F15, F17-F29, and can provide the best and consistent solutions on F6, F11, F13-F15, F18-20, F28.

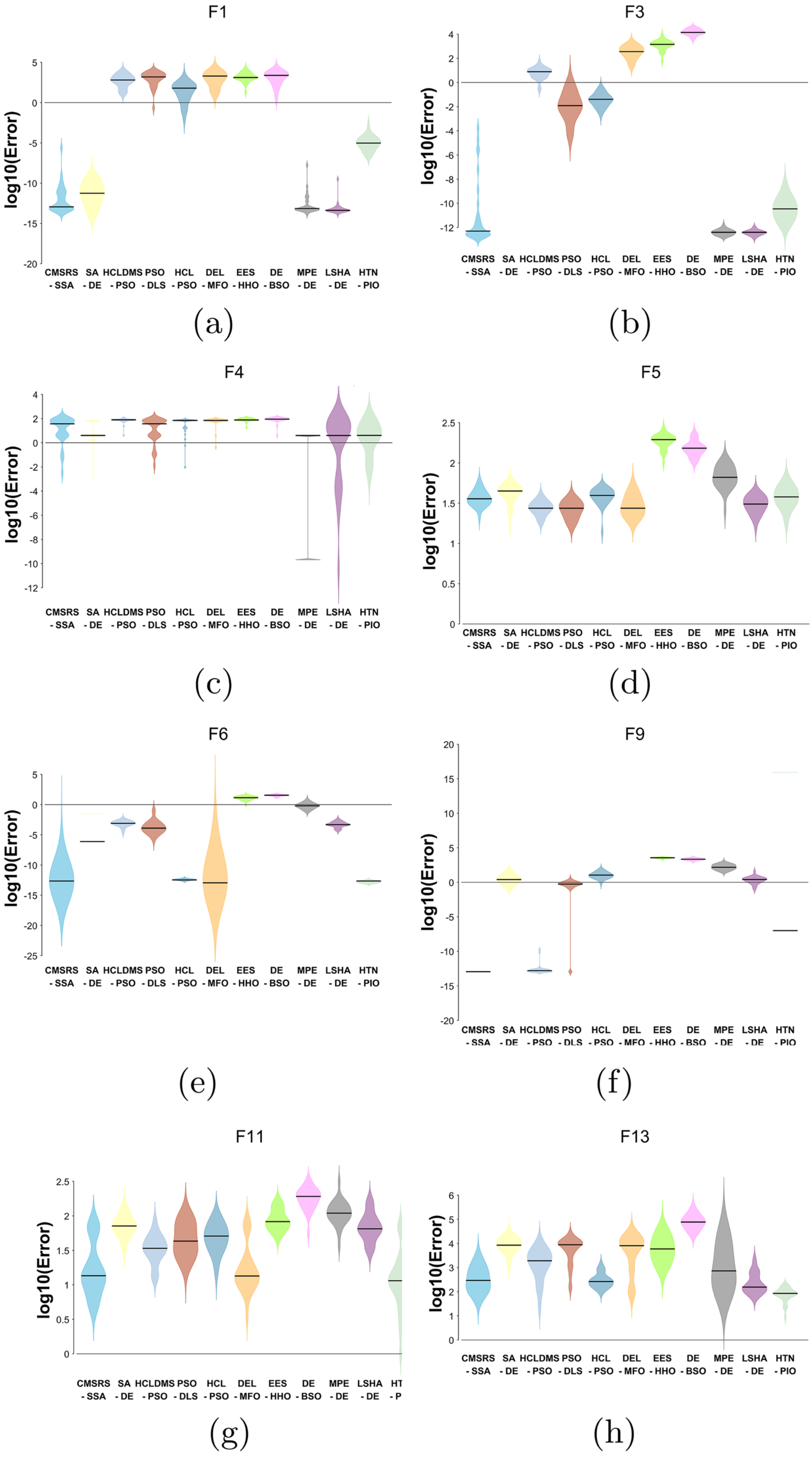
Figure 9 shows the statistical distribution of 50-D error values of 30 runs obtained by 11 advanced algorithms. For most functions, the proposed HTNPIO can provide promising solutions compared with the applied algorithms. Specifically, HTNPIO shows stable and competitive performance on F6, F7, F11, F13-F17, F19-F20, F21-F23, F26-F29, and can provide the best and consistent solutions on F14-F15, F19-20, F26.

6.6 Convergence performance analysis

Convergence curves of LFPIO, HCLPIO, LSHADE, CMSRSSSA, SADE, PSODLS, HCLPSO and HTNPIO on the $30-D$ and $50-D$ CEC2017 problems are plotted in Figs. 10 and 11, respectively. The convergence graphs are plotted by calculating an average of fitness value at a specific number of iteration.

For $30-D$ problems, HTNPIO gains the first rank on F11, F13, F14-F15, F18-F20 and F28. In these functions, the convergence speed of HTNPIO is fast. Though LSHADE and SADE converges faster than HTNPIO in the early stage, HTNPIO can achieve the most accurate in the latter stage. For F20, the fitness curve of HTNPIO is not converging fast ,however, it continues to converge, and finally achieves the best solution.

Fig. 8 Violin plots for the log_{10} error values obtained by all the algorithms (CMSRSSSA, DELMFO, SADE, HCLDMSPSO, PSODLS, HCLPSO, DEBSO, EESHHO, MPEDE, LSHADE and HTNPIO, respectively) over 30 runs for 30 dimensions benchmark functions

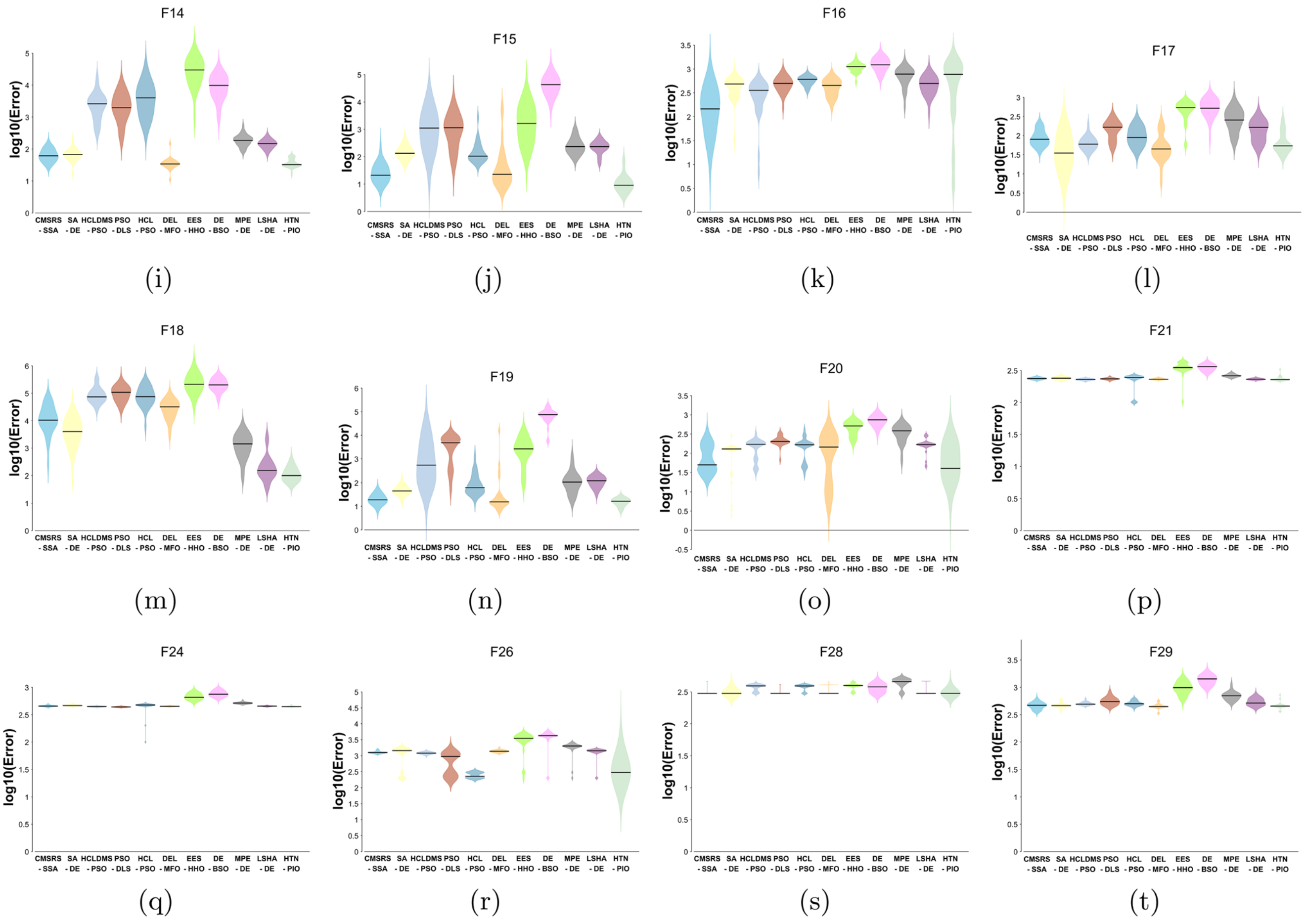


(a) (b)

(c) (d)

(e) (f)

(g) (h)



(i) (j) (k) (l)

(m) (n) (o) (p)

(q) (r) (s) (t)

Fig. 8 (continued)

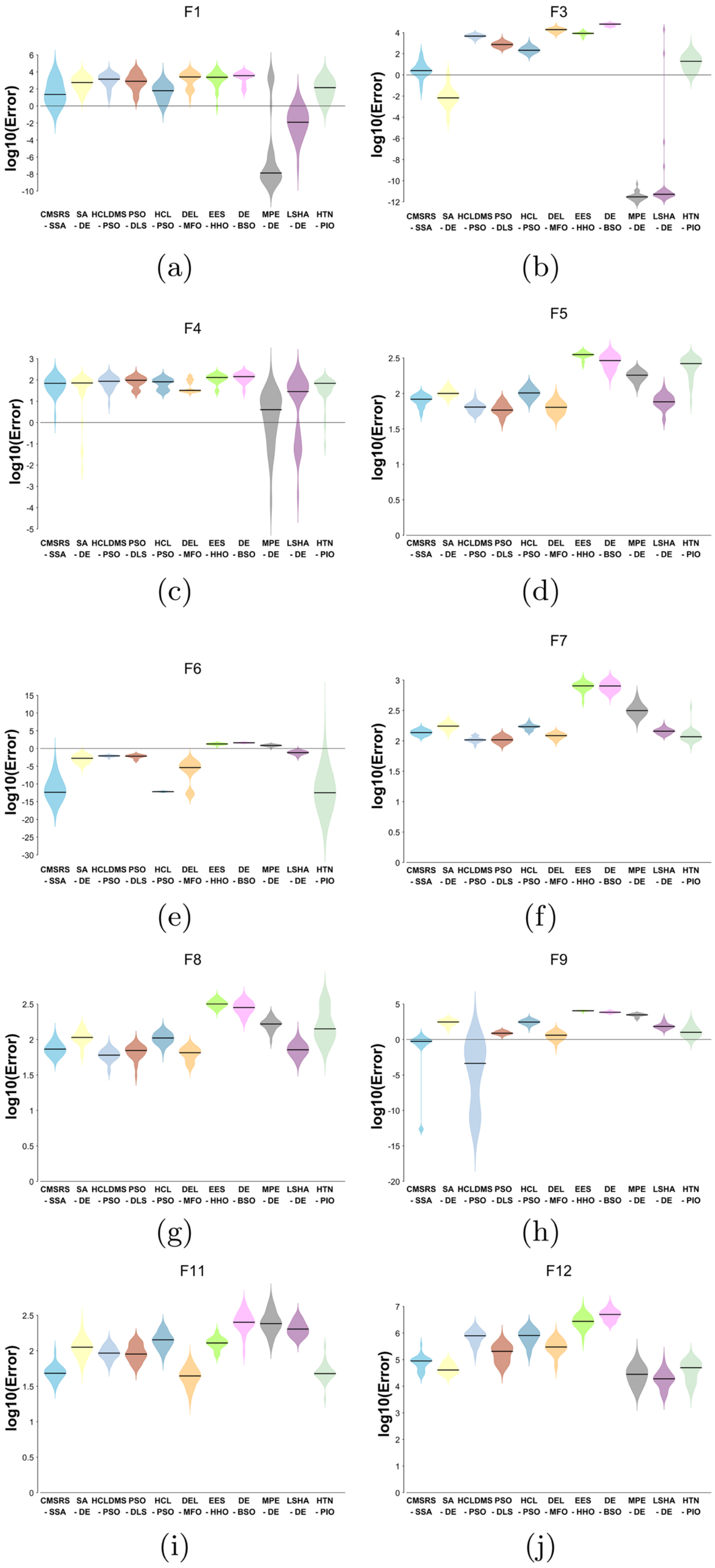
For 30 − D problems, HTNPIO gains the first rank on F14-F15, F19-F20, F22 and F26. Specifically, from the figures, HTNPIO gets a huge advantage on composition problems F22 and F26. What's more, HTNPIO has the fastest convergence rate on these problems compared with other algorithms.

From above, HTNPIO has an outstanding convergence performance. This is mainly attributed to the effective balance of global exploration and local exploitation. More specifically, SMS and LMS contribute to global exploration, and ELS focuses on local exploitation. Further, LP mechanism effectively balance the exploration and exploitation.

6.7 Impact of different ψ values on the performance of HTNPIO

ψ determines how often different mutation strategies are performed, which is of vital importance for HTNPIO. In this section, we compare HTNPIO with different ψ values on the 30 − D problems in CEC2017 to verify the impact of ψ on the performance of HTNPIO. The selected value of ψ are 0.2, 0.3, 0.4, 0.5, 0.55 (applied in this paper), 0.6, 0.7, 0.8. In this experiment, except for ψ, other parameters are kept the same as those in Table 2. What's more, the population size and maximum number of function evaluations are still set as 30 and 300000, respectively. The maximum number of iterations is 5000 (both targets and pigeons are evaluated in an iteration, separately). The results are comparatively shown in Table 7.

In Table 7, HTNPIO with ψ=0.4 obtains the first overall rank, and HTNPIO with ψ=0.2, 0.3, 0.5, 0.55, 0.6 provide better performance than with ψ=0.7 and 0.8. Though HTNPIO with ψ=0.55 is not ranked first, it is applied to make comparisons with other algorithms in CEC2017 test set and real-world optimization problems. This is because HTNPIO shows comprehensive performance and can obtain good rank among different algorithms with ψ=0.55. Additionally, the performance of HTNPIO increases first and then decreases with the increase of ψ value. This demonstrates that just using "$DE/rand/1/bin$" or "$DE/rand/2/bin$" alone cannot provide better performance for HTNPIO, and only by combining the two mutation strategies properly, HTNPIO achieves better performance.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

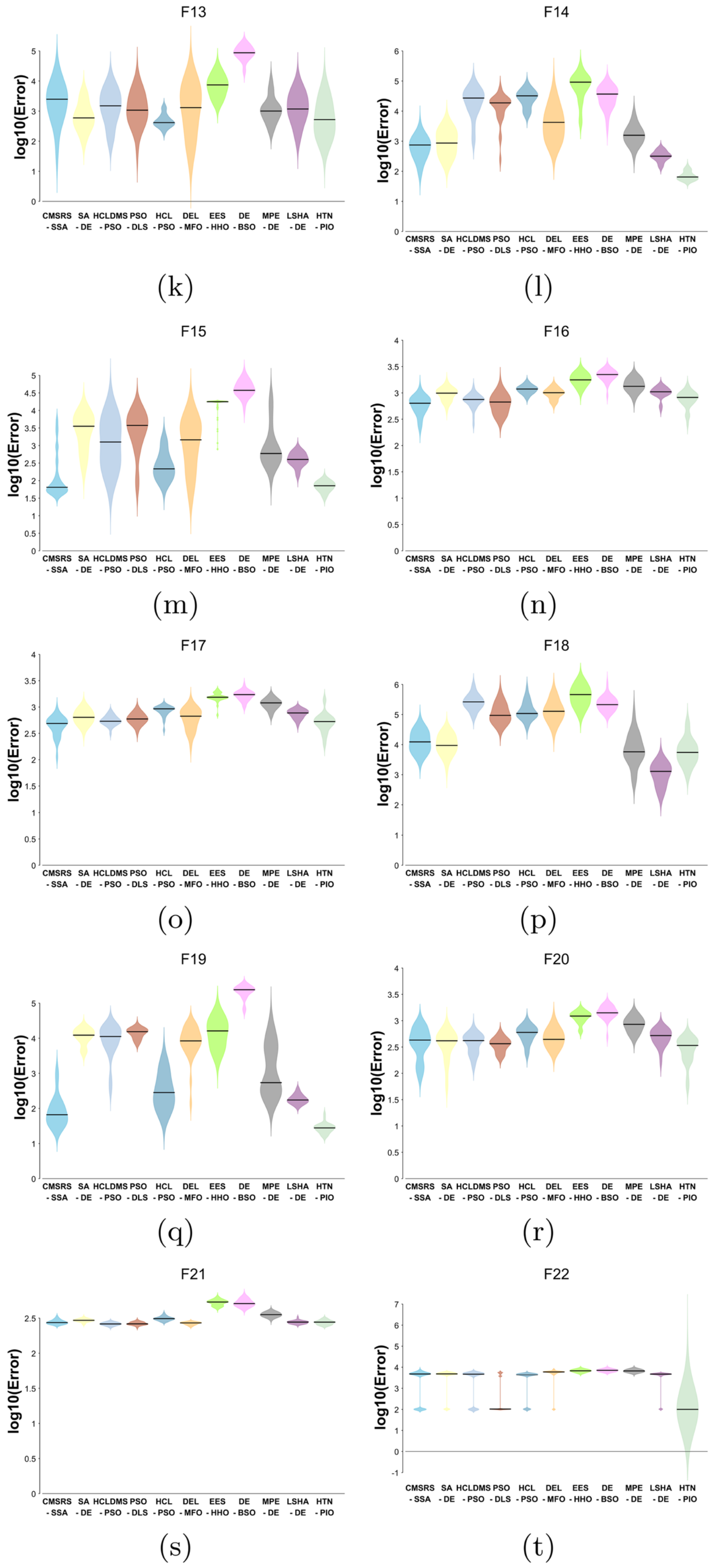
(h)

(i)

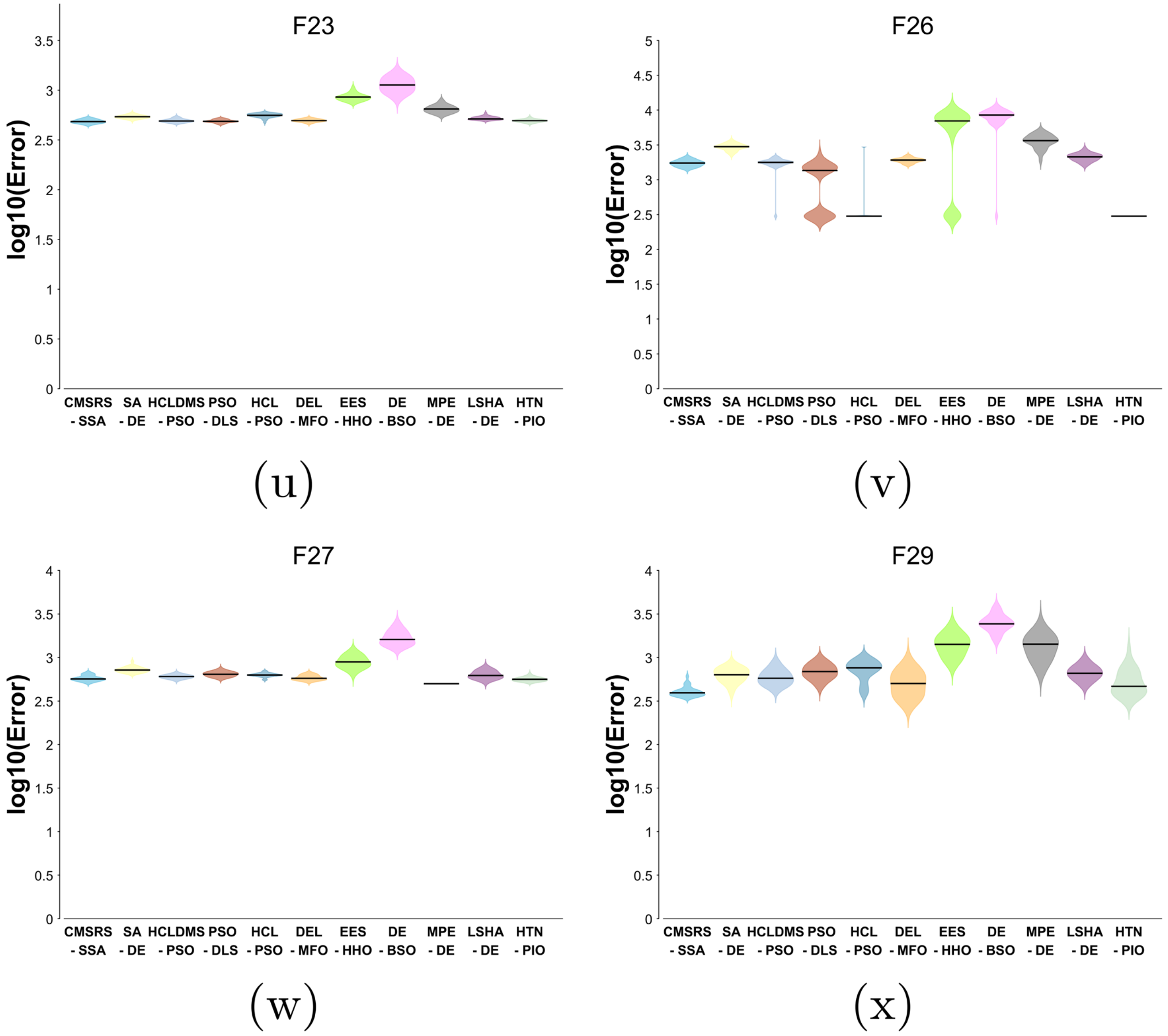
(j)

Fig. 9 Violin plots for the log_{10} error values obtained by all the algorithms (CMSRSSSA, DELMFO, SADE, HCLDMSPSO, PSODLS, HCLPSO, DEBSO, EESHHO, MPEDE, LSHADE and HTNPIO, respectively) over 30 runs for 50 dimensions benchmark functions

Fig. 9 (continued)



(k) (l) (m) (n) (o) (p) (q) (r) (s) (t)



(u)

(v)

(w)

(x)

Fig. 9 (continued)

7 Experimental results on real-world optimization problems

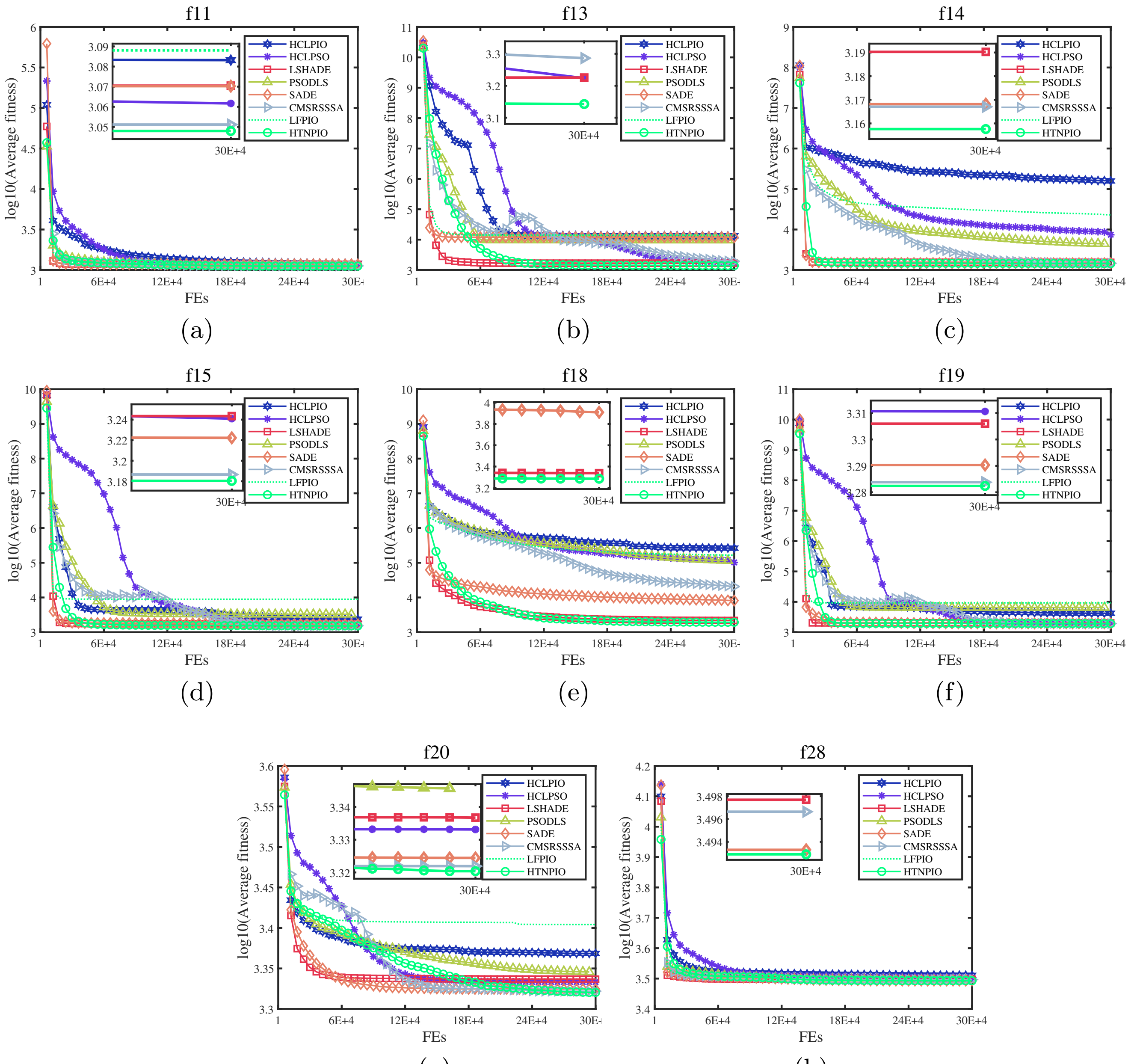
In this part, 5 real-world optimization problems are chosen from the reported literatures to further examine the ability of HTNPIO to solve practical engineering problems. The five real-world problems involve 3 different static economic load dispatch (ELD) instances [13], a hydrothermal scheduling problem [13], and a PV model parameter estimation problem [72]. Five algorithms with superior performance (LSHADE, CMSRSSSA, PSODLS, SADE, and DELMFO) and two advanced PIO algorithms (LFPIO and HCLPIO) in CEC2017 test set are applied to make comparisons with HTNPIO.

7.1 Static economic load dispatch (ELD) instances

Three static economic load dispatch (ELD) problems are chosen from the IEEE CEC2011 real-world benchmark, which are defined as follows:

$$Minimize : F = \sum_{i=1}^{N_G} f_i(P_i) \tag{34}$$

where $f_i(P_i) = a_i P_i^2 + b_i P_i + c_i, i = 1, 2, 3, \cdots, N_G$ is the cost function of the i-th generating unit, and a_i, b_i, c_i are cost coefficients. Moreover P_i is the real power output of the i-th generator. N_G is the number of online generating



(a) (b) (c)

(d) (e) (f)

(g) (h)

Fig. 10 Convergence graphs for the log_{10} mean fitness values obtained by selected algorithms over 30 runs for 30 dimensions benchmark functions

units to be dispatched. Additionally, the cost function for unit with value point loading effect is as follows:

$$f_i(P_i) = a_i P_i^2 + b_i P_i + c_i + abs(e_i sin(f_i(P_i^m in - P_i))) \quad (35)$$

where e_i and f_i are the cost coefficients corresponding to valve point loading effect. What's more, ELD problems are also constrained by energy balance, which are discussed as follows:

Power balance constraints:

$$\sum_{i=1}^{N_G} P_i = P_D + P_L \quad (36)$$

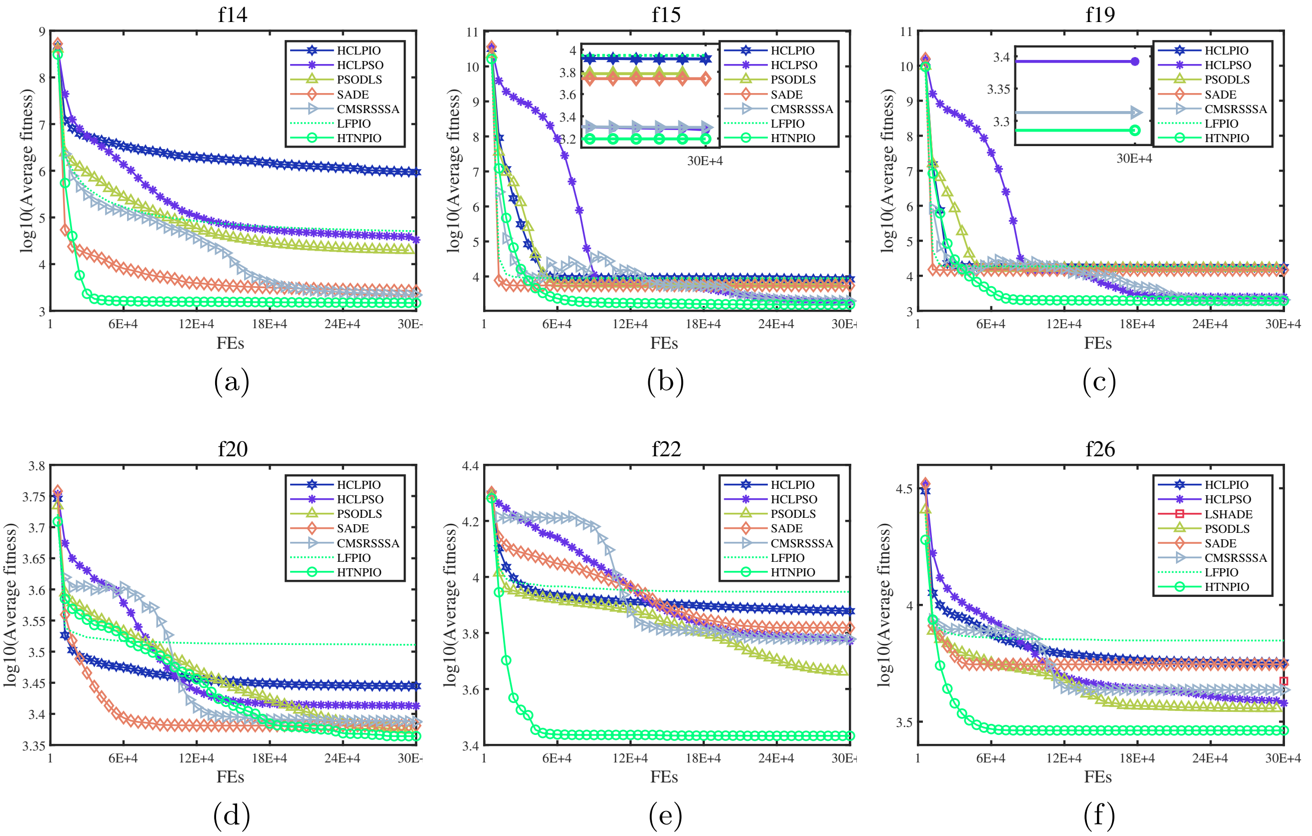
Generator constraints:

$$P_i^{\min} \geq P_i \geq P_i^{\max} \quad (37)$$

Generator constraints: where P_D and P_L represent total system loads and losses, respectively. $P_i^{\min}$ and $P_i^{\max}$ are lower and upper bounds for power outputs of the i-th generating unit.

Ramp rate limits:

$$P_i - P_i^{t-1} \leq UR_i, \quad power \quad generation \quad increases. \quad (38)$$



(a) (b) (c)

(d) (e) (f)

Fig. 11 Convergence graphs for the log_{10} (mean fitness values) obtained by selected algorithms over 30 runs for 50 dimensions benchmark functions

$$-P_i + P_i^{t-1} \leq DR_i, \quad power \quad generation \quad dcreases. \tag{39}$$

where P_i^{t-1} is the power generation of the i-th unit at previous hour, UR_i and DR_i represent the upper and lower ramp rate limits. Further, the inclusion of ramp rate limits modifies the generator operation constraints is calculated by:

$$\max(P_i^{\min}, UR_i - P_i) \leq P_i \leq min(P_i^{\max}, P_i^{t-1} - DR_i) \tag{40}$$

Prohibited operating zones

$$P_i \leq P_a^{PZ} and P_i \geq P_b^{PZ} \tag{41}$$

where P_a^{PZ} and P_b^{PZ} are the lower and upper limits of a provided zone for the $i-th$ generating unit.

7.2 Hydrothermal scheduling problem

The hydrothermal scheduling problem is extremely complex because of nonlinear relationships of the decision variables, cascaded nature of hydraulic network, water carry delays and so on. Therefore, it is difficult to obtain promising optimization results utilizing traditional optimization methods. The objective function to be minimized in a hydrothermal scheduling problem is the total fuel cost of thermal units for the given short term as follows:

$$Minimize : F = \sum_{i=1}^{M} f_i(P_{Ti}) \tag{42}$$

where f_i is the cost function for thermal unit's power generation P_{Ti} at i-th interval. M is the total number of intervals. f_i is calculated as follows:

$$f_i(P_{Ti}) = a_i P_{Ti}^2 + b_i P_{Ti} + abs(e_i sin(f_{ih}(P_{ih}^m in - P_{ih}))) \tag{43}$$

Demand constraint:

$$P_{Ti} + \sum_{k=1}^{N} P_{H(k,i)} = P_{D(i)} + P_{L(i)} \tag{44}$$

where $P_{H(k,i)}$ represents the power generated by the k-th hydro unit at the i-th interval. $P_{D(i)}$ and $P_{L(i)}$ represents the power demand and power loss. N is the total number of hydro units. Thermal generator constraint:

$$P_T^{\min} \geq P_{Ti} \geq P_T^{\max} \tag{45}$$

Table 7 Comparisons of different set of ψ with other parameters unchanged

Ins.	Indicators	ψ=0.2	ψ=0.3	ψ=0.4	ψ=0.5	ψ=0.55	ψ=0.6	ψ=0.7	ψ=0.8
F1	Mean	**4.89E-07**	5.62E-07	1.31E-06	1.84E-05	7.76E-05	8.16E-05	1.33E-04	1.72E-03
	Std	**1.05E-06**	1.31E-06	2.34E-06	5.95E-05	2.87E-04	1.75E-04	2.76E-04	5.45E-03
	Rank	**1(-)**	2(-)	3(-)	4(-)	5	6(≈)	7(≈)	8(+)
F3	Mean	2.49E-12	9.89E-13	4.61E-12	1.17E-12	3.27E-10	**9.64E-13**	1.10E-12	1.87E-12
	Std	4.45E-12	1.38E-12	1.82E-11	3.43E-12	8.90E-10	**1.70E-12**	2.68E-12	7.49E-12
	Rank	6(-)	2(-)	7(-)	4(-)	8	**1(-)**	3(-)	5(-)
F4	Mean	**3.01E+00**	1.06E+01	9.82E+00	1.51E+01	2.00E+01	1.86E+01	2.12E+01	4.65E+01
	Std	**1.84E+00**	2.18E+01	1.82E+01	2.80E+01	2.82E+01	2.86E+01	3.09E+01	3.09E+01
	Rank	**1(≈)**	3(≈)	2(≈)	4(≈)	6	5(≈)	7(≈)	8(+)
F5	Mean	8.25E+01	4.60E+01	4.98E+01	6.75E+01	**3.94E+01**	4.67E+01	5.51E+01	6.65E+01
	Std	6.02E+01	1.78E+01	2.00E+01	4.81E+01	**1.23E+01**	3.34E+01	4.15E+01	5.28E+01
	Rank	8(+)	2(+)	4(+)	7(+)	**1**	3(≈)	5(≈)	6(≈)
F6	Mean	1.37E+00	3.06E-01	1.63E-13	1.67E-13	1.74E-13	1.59E-13	1.48E-13	1.63E-13
	Std	6.18E+00	1.68E+00	5.73E-14	5.77E-14	5.77E-14	5.66E-14	5.30E-14	5.73E-14
	Rank	8(≈)	7(≈)	3(≈)	5(≈)	6	2(≈)	1(≈)	3(≈)
F7	Mean	**7.36E+01**	9.67E+01	1.14E+02	1.47E+02	1.19E+02	1.21E+02	1.48E+02	1.45E+02
	Std	**2.95E+01**	4.90E+01	5.45E+01	4.32E+01	5.12E+01	5.33E+01	3.89E+01	4.61E+01
	Rank	**1(-)**	2(-)	3(≈)	7(+)	4	5(≈)	8(+)	6(≈)
F8	Mean	9.21E+01	9.50E+01	9.47E+01	7.69E+01	7.47E+01	6.53E+01	7.17E+01	8.05E+01
	Std	3.38E+01	3.23E+01	3.37E+01	3.43E+01	2.76E+01	3.22E+01	4.09E+01	5.09E+01
	Rank	6(≈)	8(≈)	7(+)	4(≈)	3	1(≈)	2(≈)	5(≈)
F9	Mean	2.69E-01	7.23E-01	2.84E-01	1.96E-01	5.57E-01	2.96E-01	3.99E-01	**1.60E-01**
	Std	4.80E-01	1.43E+00	4.30E-01	2.32E-01	1.47E+00	4.62E-01	8.76E-01	**2.57E-01**
	Rank	3(≈)	8(≈)	4(≈)	2(≈)	7	5(≈)	6(≈)	**1(≈)**
F10	Mean	**5.06E+03**	6.11E+03	6.19E+03	6.01E+03	6.19E+03	6.34E+03	6.18E+03	6.49E+03
	Std	**1.62E+03**	4.80E+02	2.68E+02	4.38E+02	3.10E+02	2.47E+02	3.49E+02	2.78E+02
	Rank	**1(-)**	3(≈)	6(≈)	2(≈)	5	7(≈)	4(≈)	8(+)

Table 7 (continued)

Ins.	Indicators	ψ=0.2	ψ=0.3	ψ=0.4	ψ=0.5	ψ=0.55	ψ=0.6	ψ=0.7	ψ=0.8
F11	Mean	2.84E+01	2.46E+01	2.50E+01	1.90E+01	1.67E+01	1.52E+01	**1.45E+01**	2.06E+01
	Std	2.62E+01	1.85E+01	2.28E+01	1.90E+01	1.85E+01	1.54E+01	**1.16E+01**	2.50E+01
	Rank	8(+)	6(+)	7(+)	4(≈)	3	2(≈)	**1**(≈)	5(≈)
F12	Mean	1.30E+04	1.98E+04	1.76E+04	1.40E+04	1.59E+04	1.07E+04	1.49E+04	1.49E+04
	Std	1.04E+04	1.80E+04	1.98E+04	1.28E+04	1.21E+04	8.13E+03	1.35E+04	1.21E+04
	Rank	2(≈)	8(≈)	7(≈)	3(≈)	6	1(≈)	5(≈)	4(≈)
F13	Mean	**6.57E+01**	6.57E+01	8.28E+01	8.06E+01	8.91E+01	8.48E+01	9.29E+01	1.04E+02
	Std	**6.15E+01**	4.65E+01	5.20E+01	3.84E+01	4.67E+01	3.78E+01	3.30E+01	3.40E+01
	Rank	**1(-)**	2(≈)	4(≈)	3(≈)	6	5(≈)	7(≈)	8(≈)
F14	Mean	**2.94E+01**	3.22E+01	3.05E+01	3.34E+01	3.75E+01	3.99E+01	4.85E+01	6.25E+01
	Std	**1.09E+01**	9.79E+00	8.12E+00	1.28E+01	1.20E+01	1.62E+01	1.67E+01	9.69E+00
	Rank	**1(-)**	3(-)	2(≈)	4(≈)	5	6(≈)	7(+)	8(+)
F15	Mean	1.33E+01	1.23E+01	**1.06E+01**	1.12E+01	1.48E+01	1.52E+01	2.22E+01	3.04E+01
	Std	4.80E+00	6.72E+00	**4.11E+00**	6.97E+00	1.80E+01	1.41E+01	1.43E+01	1.46E+01
	Rank	4(≈)	3(≈)	**1**(≈)	2(≈)	5	6(≈)	7(+)	8(+)
F16	Mean	**4.89E+02**	6.19E+02	6.75E+02	5.09E+02	6.87E+02	7.27E+02	7.44E+02	8.08E+02
	Std	**3.77E+02**	4.46E+02	4.22E+02	3.79E+02	5.07E+02	4.14E+02	5.27E+02	4.84E+02
	Rank	**1**(≈)	3(-)	4(≈)	2(≈)	5	6(≈)	7(≈)	8(≈)
F17	Mean	5.11E+01	**4.65E+01**	7.87E+01	5.01E+01	7.95E+01	7.07E+01	9.70E+01	1.10E+02
	Std	3.54E+01	**3.40E+01**	5.54E+01	3.11E+01	6.44E+01	5.90E+01	7.42E+01	7.85E+01
	Rank	3(-)	**1**(≈)	5(≈)	2(-)	6	4(≈)	7(≈)	8(≈)
F18	Mean	4.07E+02	2.51E+02	1.35E+02	1.59E+02	1.31E+02	9.91E+01	7.67E+01	**5.48E+01**
	Std	3.18E+02	2.11E+02	1.01E+02	1.65E+02	8.60E+01	5.93E+01	6.46E+01	**9.29E+00**
	Rank	8(+)	7(+)	5(≈)	6(≈)	4	3(-)	2(-)	**1(-)**
F19	Mean	**1.08E+01**	1.13E+01	1.21E+01	1.53E+01	1.60E+01	1.91E+01	2.44E+01	2.84E+01
	Std	**3.77E+00**	5.54E+00	5.63E+00	6.73E+00	5.66E+00	7.97E+00	6.22E+00	4.15E+00
	Rank	**1(-)**	2(-)	3(-)	4(≈)	5	6(≈)	7(+)	8(+)
F20	Mean	1.17E+02	9.98E+01	9.90E+01	9.20E+01	9.12E+01	**8.52E+01**	9.21E+01	1.47E+02
	Std	5.17E+01	6.09E+01	6.23E+01	7.92E+01	7.67E+01	**6.39E+01**	1.04E+02	1.47E+02
	Rank	7(≈)	6(≈)	5(≈)	3(≈)	2	**1**(≈)	4(≈)	8(≈)

Table 7 (continued)

Ins.	Indicators	ψ=0.2	ψ=0.3	ψ=0.4	ψ=0.5	ψ=0.55	ψ=0.6	ψ=0.7	ψ=0.8
F21	Mean	2.36E+02	2.37E+02	2.37E+02	2.36E+02	2.34E+02	**2.33E+02**	2.43E+02	2.43E+02
	Std	2.99E+01	1.24E+01	1.59E+01	2.50E+01	2.12E+01	**2.77E+01**	3.97E+01	4.69E+01
	Rank	3(≈)	6(≈)	5(≈)	4(≈)	2	**1(≈)**	8(≈)	7(≈)
F22	Mean	1.00E+02	1.00E+02	**1.00E+02**	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02
	Std	4.49E-01	1.45E-13	**0.00E+00**	1.39E-13	1.45E-13	1.15E-13	1.45E-13	8.30E-14
	Rank	8(≈)	5(≈)	**1(≈)**	4(≈)	5	3(≈)	5(≈)	2(≈)
F23	Mean	3.83E+02	3.84E+02	3.86E+02	3.81E+02	3.82E+02	**3.79E+02**	4.01E+02	3.91E+02
	Std	1.16E+01	1.08E+01	2.45E+01	2.04E+01	2.49E+01	**2.79E+01**	4.89E+01	4.48E+01
	Rank	4(≈)	5(≈)	6(≈)	2(≈)	3	**1(≈)**	8(≈)	7(≈)
F24	Mean	4.57E+02	4.54E+02	4.51E+02	4.52E+02	4.47E+02	**4.46E+02**	4.56E+02	4.61E+02
	Std	1.19E+01	1.13E+01	1.13E+01	1.94E+01	9.56E+00	**1.83E+01**	3.96E+01	4.03E+01
	Rank	7(+)	5(+)	3(≈)	4(≈)	2	**1(≈)**	6(≈)	8(≈)
F25	Mean	3.86E+02	3.86E+02	**3.85E+02**	3.87E+02	3.86E+02	3.86E+02	3.86E+02	3.86E+02
	Std	1.98E+00	1.89E+00	**1.76E+00**	3.04E+00	2.01E+00	2.61E+00	1.68E+00	1.58E+00
	Rank	6(≈)	4(+)	**1(≈)**	8(+)	3	7(≈)	2(≈)	5(≈)
F26	Mean	3.52E+02	3.95E+02	**2.77E+02**	3.10E+02	4.48E+02	3.60E+02	3.31E+02	3.13E+02
	Std	2.67E+02	3.15E+02	**4.30E+01**	1.94E+02	4.76E+02	2.43E+02	2.28E+02	1.69E+02
	Rank	5(≈)	7(≈)	**1(-)**	2(-)	8	6(≈)	4(≈)	3(≈)
F27	Mean	5.05E+02	5.04E+02	**5.03E+02**	5.04E+02	5.03E+02	5.05E+02	5.03E+02	5.05E+02
	Std	6.16E+00	5.55E+00	**6.24E+00**	8.54E+00	6.78E+00	5.71E+00	6.39E+00	9.21E+00
	Rank	6(≈)	5(≈)	**1(≈)**	4(≈)	2	7(≈)	3(≈)	8(≈)
F28	Mean	3.00E+02	3.00E+02	3.00E+02	3.07E+02	3.11E+02	3.03E+02	**3.00E+02**	3.09E+02
	Std	2.35E-13	1.79E-13	1.39E-13	2.62E+01	3.37E+01	1.89E+01	**1.11E-13**	3.42E+01
	Rank	4(≈)	3(≈)	2(-)	6(≈)	8	5(≈)	**1(≈)**	7(≈)
F29	Mean	4.55E+02	4.75E+02	**4.54E+02**	4.67E+02	4.64E+02	4.72E+02	4.97E+02	5.27E+02
	Std	4.06E+01	8.10E+01	**2.97E+01**	3.75E+01	6.52E+01	5.79E+01	8.74E+01	1.12E+02
	Rank	2(≈)	6(≈)	**1(≈)**	4(≈)	3	5(≈)	7(≈)	8(+)
F30	Mean	3.83E+03	3.37E+03	3.21E+03	3.40E+03	**3.01E+03**	3.45E+03	3.51E+03	3.57E+03
	Std	1.43E+03	1.04E+03	7.75E+02	9.63E+02	**6.25E+02**	9.13E+02	1.08E+03	9.37E+02
	Rank	8(+)	3(≈)	2(≈)	4(≈)	**1**	5(+)	6(+)	7(+)
+/≈/-		5/16/8	5/18/6	3/21/5	3/22/4	/	1/26/2	5/22/3	8/19/3
Average rank		4.39	4.28	**3.62**	4.14	4.45	4.00	5.07	6.14
Final rank		4	4	**1**	2	6	3	7	8

Hydro generator constraint:

$$P_{H(k)}^{\min} \geq P_{H(k,i)} \geq P_{H(k)}^{\max} \tag{46}$$

Reservoir capacity limit:

$$V_{(k)}^{\min} \geq V_{(k,i)} \geq V_{(k)}^{\max} \tag{47}$$

where V presents the volume of each reservoir's storages.

Water discharge constraint:

$$Q_{(k)}^{\min} \geq Q_{(k,i)} \geq Q_{(k)}^{\max} \tag{48}$$

Hydraulic continuity constraint:

$$\begin{aligned} V_{(k,i+1)} = V_{(k,i)} &+ \sum_{j=\Omega_{(k)}} \left(Q_{(j,i-\tau)} + S_{(j,i-\tau)}\right) \\ &- Q_{(k,i)} - S_{(k,i)} + R_{(k,i)} \end{aligned} \tag{49}$$

Hydro power generation equation:

$$\begin{aligned} P_{H(k,i)} = & \; c_{(1,k)} V_{(k,i)}^2 + c_{(2,k)} Q_{(k,i)}^2 + c_{(2,k)} \left(V_{(k,i)} Q_{(k,i)}\right) \\ & + c_{(4,k)} V_{(k,i)} + c_{(5,k)} Q_{(k,i)} + c_{(6,k)} \end{aligned} \tag{50}$$

where $c_{(1-6,k)}$ denote constant coefficients of the k-th reservoir.

7.3 Parameter estimation of photovoltaic (PV) models

Estimation of parameters of PV models can be treated as an optimization process with specific constraints. In this issue, three models including single diode model, double diode model, and PV model are adopted. The equivalent circuit for the three models are as Fig. 12.

In the three models, the parameters to be optimized are:

$$X = [I_{ph}, I_{SD}, R_s, R_{sh}, n] \tag{51}$$

where their bounds are shown in Table 8.

The objective function for estimating the PV model's parameter is the root mean square error (RMSE). The error is the difference between all pairs of measured and calculated current results. Precisely, RMSE is expressed as follows:

$$RMSE(X) = \sqrt{\frac{\sum_{k=1}^{N} Fe\left(V_L, I_L, X\right)}{N}} \tag{52}$$

7.4 Comparison results of real-world optimization problems

The optimization results of HTNPIO and 5 other algorithms on 5 real-world optimization problems are shown in the Table 9. The best, worst, mean and standard deviation (STD) value of each problem's objective function are shown in the table. Notably, the best and the secondary entry of each indicator is highlighted in bold.

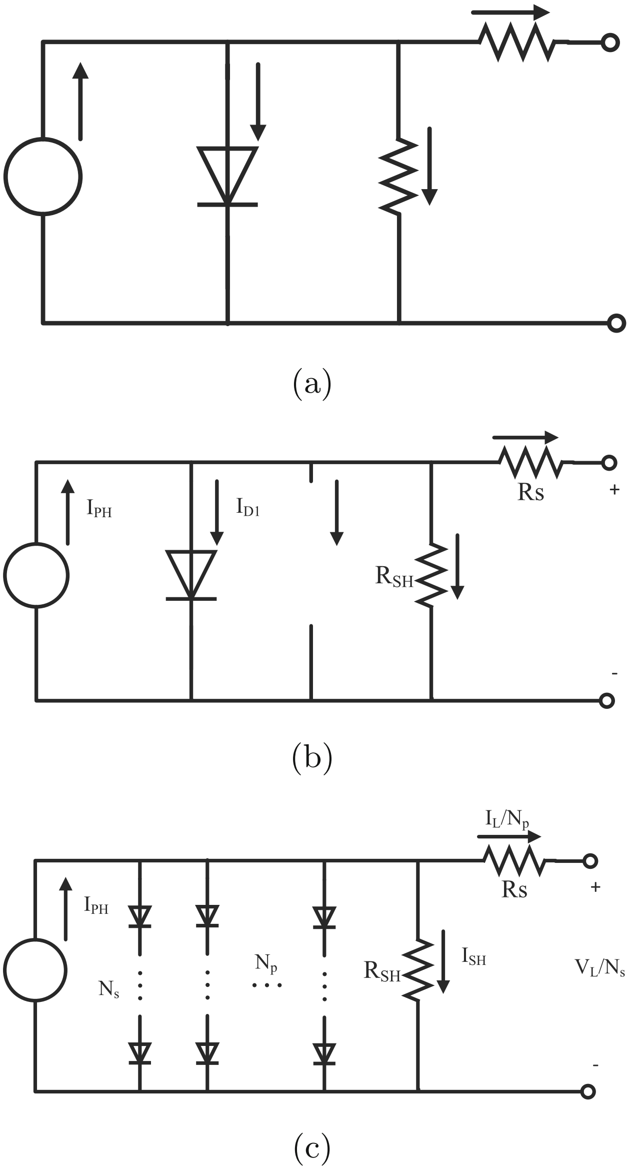


Fig. 12 Equivalent circuits of single diode, double diode and PV models

For instance 1, the proposed HTNPIO beats all the competitors in the mean, STD, and worst value.

For instance 2, HTNPIO obtained the first rank in STD value, and provided the second best results in the worst, mean and best value, following LSHADE.

For instance 3, HTNPIO acquires the best STD and maximum objective function value of the 140-D ELD problem, and beat all competitors except LSHADE in mean and best value. However, the results provided by HTNPIO and LSHADE differ by less than 1%, which is acceptable.

For the 96-D hydrothermal scheduling problem, HTNPIO beat all the advanced compared algorithms with huge advance.

Table 8 Upper and lower bounds of parameters of 3 different PV models

Parameter	Single diode & double diode		PV module	
	Lower bound	Upper bound	Lower bound	Upper bound
IPH (A)	0	1	0	2
ID, ID1, ID2 (μA)	0	1	0	50
Rs (Ω)	0	0.5	0	2
RSH (Ω)	0	100	0	2000
Ns, Np	1	2	1	50

In the fifth instance, for single-diode model, HTNPIO, LSHADE and SaDE can all acquire similar and stable RMSE value in independent 30 runs; for PV model, HTNPIO, CMSRSSSA, LSHADE, and SaDE can similarly obtain the best RMSE results.

In summary, HTNPIO can always provide first-rate results and rank first in all real-world optimization problems, which shows extremely competitive performance compared with other state-of-the-art algorithms.

Additionally, computation time is an important factor of an algorithm, especially when dealing with real-world problems. Figure 13 displays the time taken by each algorithm to handle different real-world optimization problems. Moreover, the time for each of the 30 runs is given. In the figures, the x-axis represents the time each algorithm ran, the y-axis represents the number of runs. It is observed that the computational time of HTNPIO is a little more than that of HCLPIO, CMSRSSSA and LFPIO, but much less than that of SaDE in the first two case, and time for LSHADE and HTNPIO is similar. For the third instance, HTNPIO consumes less time than CMSRSSSA and SaDE, and similar to HCLPIO. For instance 4, the time consumed by HTNPIO is far less than that of LSHADE and SaDE, and similar to that of HCLPIO and PSODLS. What's more, time for instance 5 of HTNPIO is almost the least, however, HTNPIO can still provide the best solutions in the problems. Comprehensively, the time taken by HTNPIO to process real-world problems is always stable. Though the computational time of HTNPIO is not always ranked first, it is sitll acceptable because HTNPIO can provide promising and extremely competitive solutions for each instance.

8 Discussion

The conventional PIO algorithm utilizes map-compass operator and landmark operator to achieve rapid convergence. However, the conventional PIO is still trapped into local optima especially when solving complicated shifted rotated and non-continuous composition optimization problems. Major reasons are listed as follows:

First, map-compass operator makes pigeons only learn from a global best individual in the current iteration, leading to insufficient information interchanges among the whole population. Such interchanges of information makes the pigeons susceptible to the loss of diversity, such as losing some potential pigeons. In this case, the pigeons are easily stuck into local optima and it is hard for them to jump out of the local optima.

Second, landmark operator forces a reduction in the number of pigeons in the later iteration, accelerating the lack of population diversity in the later iteration. In each generation, the population size decreases by 1/2, causing insufficient examplars to guide the search pigeon, which limits the ability to jump out of local optima. This can be clearly demonstrated by Figs. 5, 6 and 7 in Section 5, where the group in PIO accelerate convergence and then came together in 10-th iteration. However, the population in PIO is concentrated around the current best point (x=-10,y=25) and loses the global search ability, so it cannot jump out of the local optimum in the 20-th iteration. This's the downside when PIO handles problems. The same conclusion can be drawn from the Fig. 7.

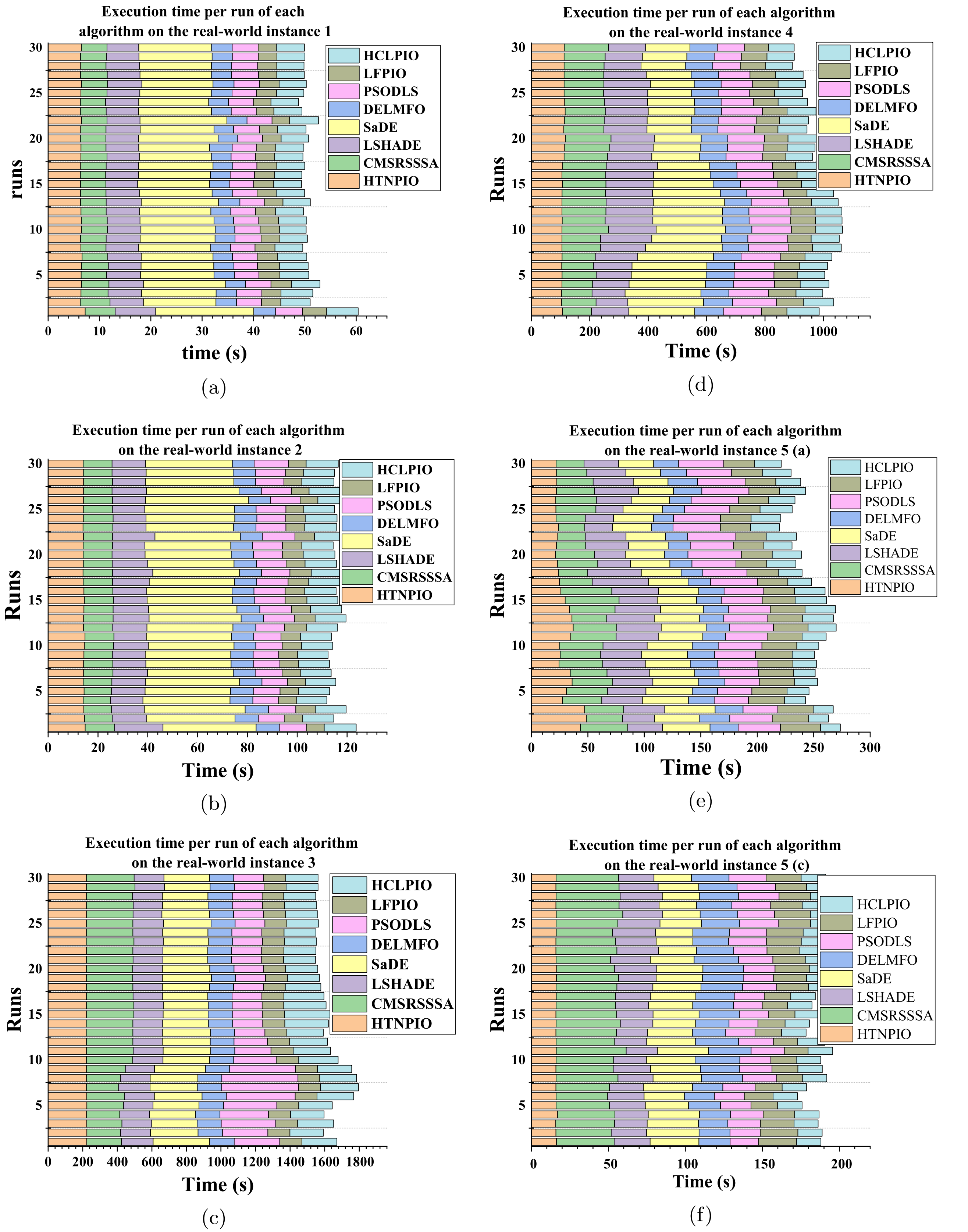
To address the above problems, although most presented PIO variants [8, 27, 31, 34, 70, 71, 75] adopt various strategies to improve the propoties of PIO, those works seldom consider how to refrain the drawvacks of the tedious learning strategy in the map-compass operator. Therefore, most existing PIOs are also easy to got caught into the local optimum when solving out the complex problems such as IEEE CEC2017 test suite.

To address the mentioned issues, we have proposed a HTNPIO algorithm, which adopts SMS and LMS to generate high-level targets and efficiently navigate the search of pigeons, and adopts ELS to make full use of the brilliant convergence capability of PIO. What's more, the LMS-ELS probability mechanism is proposed to balance the exploitation and exploration of HTNPIO.

First, HTNPIO's learning strategies are diversified, not as single as PIO, therefore, HTNPIO effectively guarantees the diversity of the population, so that it is more possible to find optimal solutions. Second, HTNPIO inherits the

Table 9 Comparisons of optimal results gained by 6 advanced algorithms for 5 real-world problems (the best and second entry is highlighted in bold)

Real-world optimization ins.		Indicators	HTNPIO	CMSRSSSA	LSHADE	PSODLS	LFPIO	HCLPIO	SaDE	DELMFO
Instance 1	15-D ELD	Worst	**3.2911E+04**	**3.2974E+04**	3.3035E+04	3.3228E+04	3.3392E+04	3.3225E+04	3.3004E+04	3.3014E+04
		Mean	**3.2839E+04**	3.2881E+04	**3.2862E+04**	3.3062E+04	3.3144E+04	3.3124E+04	3.2908E+04	3.2927E+04
	instance	Best	3.2778E+04	**3.2747E+04**	**3.2769E+04**	3.2981E+04	3.2933E+04	3.2920E+04	3.2829E+04	3.2873E+04
		STD	**3.2817E+01**	4.8579E+01	7.4062E+01	5.0149E+01	9.3068E+01	7.0700E+01	4.5523E+01	**3.4920E+01**
Instance 2	40-D ELD	Worst	**1.3710E+05**	1.4486E+05	**1.3521E+05**	1.4645E+05	1.5213E+05	6.0201E+08	1.3979E+05	1.3872E+05
		Mean	**1.3183E+05**	1.3526E+05	**1.2882E+05**	1.3850E+05	1.4503E+05	5.9894E+08	1.3426E+05	1.3437E+05
	instance	Best	**1.2913E+05**	1.2956E+05	**1.2655E+05**	1.3353E+05	1.3672E+05	5.9748E+08	1.3051E+05	1.3131E+05
		STD	**1.7194E+03**	3.6322E+03	**1.7245E+03**	2.9343E+03	4.5515E+03	9.1392E+05	2.2910E+03	1.9393E+03
Instance 3	140-D ELD	Worst	**1.9410E+06**	1.9744E+06	1.9786E+06	1.9683E+06	3.8279E+06	7.7900E+11	2.2713E+08	2.0027E+06
		Mean	**1.9184E+06**	1.9482E+06	**1.9049E+06**	1.9428E+06	2.6303E+06	7.7814E+11	9.7227E+06	1.9551E+06
	instance	Best	**1.8862E+06**	1.9172E+06	**1.8767E+06**	1.9161E+06	1.9296E+06	7.7749E+11	1.9153E+06	1.9220E+06
		STD	**1.2396E+04**	1.4457E+04	2.4664E+04	**1.2935E+04**	5.5255E+05	4.0569E+08	4.1072E+07	2.2329E+04
Instance 4	96-D hydro	Worst	**9.4454E+05**	9.5162E+05	9.5379E+00	**9.4570E+05**	4.0535E+06	1.1911E+06	9.4696E+05	1.1891E+06
		Mean	**9.4081E+05**	9.4603E+05	9.4573E+05	**9.4261E+05**	1.2236E+06	9.8493E+05	9.4304E+05	9.6748E+05
	instance	Best	**9.3845E+05**	**9.3777E+05**	9.4014E+00	9.3943E+05	9.4171E+05	9.3919E+05	9.3893E+05	9.4101E+05
		STD	**1.5354E+03**	3.1307E+03	3.0120E+03	1.8292E+03	6.0451E+05	7.6190E+04	**2.1393E+03**	5.4365E+04
Instance 5	Single	Worst	**9.8602E-04**	9.9258E-04	**9.8602E-04**	1.7562E-03	2.5822E-03	2.3053E-03	**9.8602E-04**	1.0468E-03
	diode	Mean	**9.8602E-04**	9.8676E-04	**9.8602E-04**	1.2946E-03	1.5688E-03	1.3754E-03	**9.8602E-04**	9.9492E-04
	model	Best	**9.8602E-04**	**9.8602E-04**	**9.8602E-04**	1.0214E-03	9.8672E-04	9.8624E-04	**9.8602E-04**	**9.8602E-04**
		STD	4.3354E-10	1.4802E-06	**3.1506E-17**	1.8709E-04	4.3541E-04	3.8378E-04	**2.0616E-17**	1.5619E-05
	Double	Worst	1.5540E-03	**1.1355E-03**	1.9384E-03	2.0109E-03	2.3350E-03	3.0179E-03	**9.8602E-04**	1.2308E-03
	diode	Mean	1.1518E-03	**1.0090E-03**	1.0243E-03	1.4953E-03	1.3802E-03	1.9104E-03	**9.8354E-04**	1.0186E-03
	model	Best	9.8475E-04	9.8251E-04	**9.8248E-04**	9.8472E-04	9.8646E-04	9.8752E-04	**9.8248E-04**	9.8344E-04
		STD	1.6756E-04	**3.8995E-05**	1.7774E-04	2.7635E-04	3.7520E-04	6.9808E-04	**1.6214E-06**	6.4186E-05
	PV-	Worst	2.4257E-03	2.4252E-03	**2.4251E-03**	2.5457E-03	1.0295E-02	4.5016E-03	**2.4251E-03**	2.4330E-03
		Mean	**2.4251E-03**	**2.4251E-03**	**2.4251E-03**	2.4431E-03	3.1931E-03	2.7246E-03	**2.4251E-03**	2.4255E-03
	model	Best	**2.4251E-03**	**2.4251E-03**	**2.4251E-03**	**2.4251E-03**	2.4262E-03	2.4272E-03	**2.4251E-03**	**2.4251E-03**
		STD	1.1214E-07	1.5234E-08	**2.2610E-17**	2.5490E-05	1.5430E-03	4.6296E-04	1.7760E-17	1.6647E-06



(a)

(d)

(b)

(e)

(c)

(f)

Fig. 13 The execution time for each run on 5 real-world optimization problems achieved by HTNPIO, SaDE, LSHADE, DELMFO, PSODLS, CMSRSSSA, LFPIO and HCLPIO

good convergence ability of PIO. What's more, HTNPIO emphasizes to balance the exploration and exploitation in each search stage, thus effectively improves the global search ability and convergence accuracy of PIO. In detailed, the advantages of each strategy are demonstrated as follows.

Selected mutation strategy (SMS) facilitates the information interaction of personal best individuals in the population, and further generates high-level targets to efficiently navigate the search behavior of pigeons. SMS employs two kinds of mutation strategies to provide random and effective information interactions between personal bests (PBs) to generate targets (Ts), as the data in Ts is diversified and high-level, SMS can acquire large exploration scopes in the solution space and contribute to enhance the global search ability. Because of the application of the two mutation strategies, SMS also employs two kinds of crossover strategies to perform dimensional update for each pigeon, which can further improve the population diversity of targets. Selection strategy is executed to maintain the high-level targets, and guarantee the whole population to develop in a better direction.

Compared with PIO's map and compass strategy, HTNPIO's levy-based map-compass strategy (LMS) has the advantage of providing more diverse learning exemplars. LMS allows pigeons to explore diverse and high-level targets, and obtain a larger search range around potential solutions in the space. What's more, levy flight model is an effective method to accomplish irregular movement with random steps, thus, the proper application of levy flight model makes the search more random, which could further intensify the global search ability of HTNPIO in the entire iteration. The advantages of SMS and LMS can be demonstrated by CEC2017 test suite. In Tables 5 and 6, HTNPIO obtains the most first ranks in multimodal and composition problems such as F11-F15, F18-F20 and F26-F28, which exhibits the superior global search capability of HTNPIO, verifying that the diversity of HTNPIO is maintained well. This can also be visualized and demonstrated by Figs. 5, 6, 7 in Section 5. In the above figures, the points (HTNPIO individuals) are scattered over a wide range in the search space, validating that HTNPIO has still maintained a good population dispersion in the 20-th iteration. So even in the last iteration, the individuals of HTNPIO are not clustered together, and even some points are scattered far from the global optimum, which provides the possibility for subsequent searches to find possible better values, which guarantees global search capability.

The advantage of ELS is that it provides fast and accurate convergence. ELS enables individuals to exploit around elites in the late iteration, which effectively accelerates convergence and gains accurate solutions. What's more, ELS also performs dimensional updates, which means that ELS is more likely to outcrop the optimal value of the potential area. The advantages of the strategy can also be demonstrated by Figs. 5 and 7 in Section 5. From Figs. 5 and 7 convergence in the later stage, however, the individuals also scatter to be able to explore potential somehow.

The advantage of a linear LMS-ELS probabilistic selection mechanism is that it can effectively balance the exploration and exploitation of HTNPIO in different stages of iteration. In early iterations, this mechanism selects most of the pigeons to perform LMS to explore potential solutions in the solution space, while in the later stage, more pigeons are forced to execute the ELS to exploit the current best solutions. By this mechanism, the algorithm tries to search for potential solutions in a wide search range in the early iterative stage, and quickly taps the optimal region in the later iterative stage to search for a more accurate solution.

Based on the above advantages, the proposed HTNPIO outperforms other five advanced PIO variations and ten state-of-the-art on both 30 and 50 dimension problems in IEEE CEC2017 test suite, and can provide extremely competitive performance on real-world optimization problems. The main numerical achievements of HTNPIO are as follows:

In CEC2017 test suite, the HTNPIO can attain solutions on F1 with $30 - D$ whose mean error is only 7.76×10^{-5}, on F3, the accuracy of the solution provided by HTNPIO is about 10^{-10}. On these two problems, HTNPIO improves the accuracy of PIO by more than 10^{10} times. HTNPIO also owes outstanding performance on F6, and the average error value from the optimal solution is only 1.74×10^{-13}, which ranks first among all compared algorithms. HTNPIO also performs best on multimodel problems F11, F13, F14, F15, with an improvement of the accuracy of the PIO variant by more than 10 times. On F18 and F19, HTNPIO could provide solutions with accuracy more than 3 times higher than other advanced algorithms. On $50 - D$ problems, HTNPIO improves the accuracy of PIO by at least 10 times on all functions. what's more, on F14 and F15, HTNPIO can provide solutions with error 7.34 and 7.18, respectively, which 5 times more accurate than LSHADE. HTNPIO's solution to the F26 has an average error value of 508 from the true optimum, which is 10 times more accurate than all other state-of-the-art algorithms. Additionally, the solutions of HTNPIO on F26 is 300, competing all competitors. Comprehensively, compared with PIO variants, HTNPIO has huge advantages. HTNPIO can get 26 firsts on 29 $30-D$ problems, and 25 first ranks on 29 $50 - D$ problems, with undoubtedly first final rank on both $30 - D$ and $50 - D$ problems. When compared to other highly competitive advanced algorithms, HTNPIO can get 9 first place on 29 $30-D$ functions, which wins the most first ranks among all algorithms. With an average rank of 3.59, HTNPIO wins the first final rank. On $50 - D$ problems, HTNPIO ranks first for 6 times, which is also the most among competitors. The average rank of HTNPIO is 3.69, which is also the best.

However, the disadvantages and limitations of HTNPIO are as follows:

First, key parameters of HTNPIO such as CR and F in the selection mutation strategy rely on experience and cannot be adaptively adjusted according to the specific problem, which limits the performance of HTNPIO on some problems. Second, SMS and LMS may cause excessive dispersion of pigeons in the search space, which is effective for multi-modal and complex problems, but may result in inaccurate results for simple or single-mode problems. For example, HTNPIO cannot provide the best solutions on unimodal problems in CEC2017 when comparing with advanced algorithms. HTNPIO also fails to provide the best solution on the 15-D ELD instance and the best mean result on double-diode model instance in the real-world optimization problems. Third, LMS or ELS is selected by a linear probability mechanism, however, the linear probability may not provide the best trade off between exploration and exploitation on some specific problems. For example, for a unimodal problem, a large number of scattered searches in early stage can lead to slow convergence and inferior accuracy within a certain number of iterations. Moreover, when HTNPIO's decentralized search does not find potential positions, and at this time, individuals in the population perform more LMS to accelerate convergence, which will reduce the possibility of jumping out of the local optimum. Therefore, to optimize and improve HTNPIO, it is necessary to consider the adaptive changes of parameters and the adaptive execution of strategies, which is also the possible direction of future work.

Additionally, from the computational complexity analysis and computational time, HTNPIO is not the fastest, however, the computational time of HTNPIO is also promising because HTNPIO can acquire rather accurate solutions. Therefore, HTNPIO has a relatively broad application outlook with short computing time, low computing cost, and strong global optimization ability. Authors recommend HTNPIO can be used for solving complex problems such as multimodal, hybrid or composition problems instead of low-dimensional simple problems. Also, HTNPIO can be utilized to solve engineering problems that have high demands on accuracy and computation time. Furthermore, HTNPIO can be improved for multi-objective optimization and fuzzy mathematical programming [61, 62].

9 Conclusion

This paper proposed a new PIO variant named HTNPIO, where three strategies, namely selective mutation strategy (SMS), levy-based map-compass strategy (LMS) and enhanced landmark strategy (ELS) are included. What's more, an LMS-ELS probability (LP) mechanism is proposed to balance the exploration and exploitation. Firstly, in SMS, differential strategy makes interactions among different personal best pigeons to generate random high-level targets. The establishment of these high-level targets expands the search scopes of the pigeons in the solution space, thereby increasing the likelihood of finding the global optima. Secondly, LMS is executed to explore potential places around these high-level targets instead of the current global best. In this process, the pigeons' search trajectory is more diverse so the chances of the pigeons finding the optimal solution are greatly increased. Thirdly, we present the ELS to accelerate convergence and improve the accuracy of solutions. Furthermore, we propose an LP mechanism to balance the exploration and exploitation in the complete iterative process. Experimental results shows that HTNPIO defeats all the competitors on $30 - D$ and $50 - D$ CEC2017 problems. Moreover, HTNPIO also exhibits first-rate performance on the 5 real-world optimization problems compared with advanced algorithms. Hence, HTNPIO has big improvements in global search capability, local exploitation ability, and steadily.

In the future, HTNPIO can be used for solving various constrained engineering problems.

Acknowledgements This work was supported by National Natural Science Fundation of China under grants number of 51507813.

Declarations

- National Natural Science Fundation of China [grants number 51507813]
- There are no conflict of interest/Competing interests.
- The data and materials are available.
- Source code of HTNPIO will be available if the paper is considered to be published.

References

1. Akay B, Karaboga D (2012) A modified artificial bee colony algorithm for real-parameter optimization. Inf Sci 192:120–142
2. Alazzam H, Sharieh A, Sabri KE (2020) A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. Expert Syst Appl 148:113–249
3. Alazzam H, Sharieh A, Sabri KE (2021) A lightweight intelligent network intrusion detection system using ocsvm and pigeon inspired optimizer. Applied Intelligence
4. Awad N, Ali M, Liang J et al (2016) Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization. NTU, Singapore, Tech Rep
5. Bäck T, Schwefel HP (1993) An overview of evolutionary algorithms for parameter optimization. Evol Comput 1:1–23
6. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. J Mach Learn Res 13:281–305

7. Biswas PP, Suganthan PN, Amaratunga GAJ (2017) Minimizing harmonic distortion in power system with optimal design of hybrid active power filter using differential evolution. Appl Soft Comput 61:486–496
8. Zhang B, Duan HB (2014) Predator-prey pigeon-inspired optimization for UAV three-dimensional path planning. In: Advances in swarm intelligence, ICSI 2014, PT II, pp 96–105
9. Chen G, Qian J, Zhang Z et al (2020) Application of modified pigeon-inspired optimization algorithm and constraint-objective sorting rule on multi-objective optimal power flow problem. Appl Soft Comput 92:106–321
10. Chen Y, Li L, Peng H et al (2018) Dynamic multi-swarm differential learning particle swarm optimizer. Swarm Evol Comput 39:209–221
11. Cheung NJ, Ding X, Shen HB (2015) A supervised particle swarm algorithm for real-parameter optimization. Appl Intell 43:825–839
12. Coello CAC, Pulido GT, Lechuga MS (2004) Handling multiple objectives with particle swarm optimization. IEEE Trans Evol Comput 8:256–279
13. Das S, Suganthan PN (2010) Problem definitions and evaluation criteria for cec 2011, competition on testing evolutionary algorithms on real world optimization problems
14. Das S, Suganthan PN (2011) Differential evolution: A survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31
15. Deb K, Pratap A, Agarwal S et al (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput 6(2):182–197
16. Derrac J, Garcia S, Molina D et al (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18
17. Dervis K, Bahriye B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Glob Optim 39(3):459–471
18. Zhang DF, Duan HB (2018) Identification for a reentry vehicle via Levy flight-based pigeon-inspired optimization. Proc Inst Mech Eng G J Aerosp Eng 232(4):626–637
19. Dorigo M, Maniezzo V, Colorni A (1996) Ant system: optimization by a colony of cooperating agents. IEEE Trans Syst Man Cybern Part B (Cybernetics) 26(1):29–41
20. Duan H, Qiao PX (2014) Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning. Int J Intell Comput Cybern 7(1):24–37
21. Duan H, Huo M, Shi Y (2020) Limit-cycle-based mutant multiobjective pigeon-inspired optimization. IEEE Trans Evol Comput 24:948–959
22. Duan H, Zhao J, Deng Y et al (2021) Dynamic discrete pigeon-inspired optimization for multi-uav cooperative search-attack mission planning. IEEE Trans Aerosp Electron Syst 57:706–720
23. Eiben AE, Smith JE (2015) From evolutionary computation to the evolution of things. Nature 521:476–482
24. Ge F, Li K, Han Y et al (2020) Path planning of uav for oilfield inspections in a three-dimensional dynamic environment with moving obstacles based on an improved pigeon-inspired optimization algorithm. Appl Intell 50:2800–2817
25. Ghosh S, Das S, Roy S et al (2012) A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization. Inf Sci 182:199–219
26. Wu GH, Mallipeddi R, Suganthan PN et al (2016) Differential evolution with multi-population based ensemble of mutation strategies. Inf Sci 329(SI):329–345
27. Tao GJ, Li Z (2018) A crossed pigeon -inspired optimization algorithm with cognitive factor. J Sichuan Univ (Med Sci Ed), pp 295–300
28. Guo P, Zhu L (2012) Ant colony optimization for continuous domains. In: 2012 8th International conference on natural computation, pp 758–762
29. Hai X, Wang Z, Feng Q et al (2019) Mobile robot adrc with an automatic parameter tuning mechanism via modified pigeon-inspired optimization. IEEE/ASME Trans Mechatronics 24:2616–2626
30. Hakli H, Uguz H (2014) A novel particle swarm optimization algorithm with levy flight. Appl Soft Comput 23:333–345
31. Duan HB, Huo MZ, Yang ZY, et al (2019) Predator-prey pigeon-inspired optimization for UAV ALS longitudinal parameters tuning. IEEE Trans Aerosp Electron Syst 55(5):2347–2358
32. HE H, Duan H (2021) A multi-strategy pigeon-inspired optimization approach to active disturbance rejection control parameters tuning for vertical take-off and landing fixed-wing uav. Chinese Journal of Aeronautics
33. Heidari AA, Pahlavani P (2017) An efficient modified grey wolf optimizer with lévy flight for optimization tasks. Appl Soft Comput 60:115–134
34. Li HH, Duan HB (2014) Bloch Quantum-behaved Pigeon-Inspired Optimization for Continuous Optimization Problems. In: 2014 IEEE Chinese guidance, navigation and control conference (CGNCC), pp 2634–2638
35. Houssein EH, Saad MR, Hashim FA et al (2020) Lévy flight distribution: A new metaheuristic algorithm for solving engineering optimization problems. Eng Appl Artif Intell 94:103,731
36. Iacca G, dos Santos Junior VC, de Melo VV (2021) An improved jaya optimization algorithm with lévy flight. Expert Syst Appl 165:113,902
37. Jiang F, He J, Tian T (2019) A clustering-based ensemble approach with improved pigeon-inspired optimization and extreme learning machine for air quality prediction. Appl Soft Comput 85:105827
38. Kadirkamanathan V, Selvarajah K, Fleming PJ (2006) Stability analysis of the particle dynamics in particle swarm optimizer. IEEE Trans Evol Comput 10:245–255
39. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN95 international conference on neural networks, pp 1942–1948
40. Khalilpourazari S, Khalilpourazary S (2019) An efficient hybrid algorithm based on water cycle and moth-flame optimization algorithms for solving numerical and constrained engineering optimization problems. Soft Comput 23(5):1699–1722
41. Li C, Niu Z, Song Z et al (2018) A double evolutionary learning moth-flame optimization for real-parameter global optimization problems. IEEE Access 6:76,700–76727
42. Li C, Li J, Chen H et al (2021) Enhanced harris hawks optimization with multi-strategy for global optimization tasks. Expert Syst Appl 185:115–499
43. Li W, Meng X, Huang Y et al (2020) Multipopulation cooperative particle swarm optimization with a mixed mutation strategy. Inf Sci 529:179–196
44. Ling YB, Zhou Y, Luo Q (2017) Lévy flight trajectory-based whale optimization algorithm for global optimization. IEEE Access 5:6168–6186
45. Liu W, Liu L, Chung IY et al (2011) Real-time particle swarm optimization based parameter identification applied to permanent magnet synchronous machine. Appl Soft Comput 11:2556–2564
46. Lynn N, Suganthan P (2015) Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. Swarm Evol Comput 24:11–24
47. Mareda T, Gaudard L, Romerio F (2017) A parametric genetic algorithm approach to assess complementary options of large scale wind-solar coupling. IEEE/CAA J Automatica Sinica 4(jas–4–2–260):260

48. Mirjalili S (2015) Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowl Based Syst 89:228–249
49. Pan JS, Tian AQ, Chu SC et al (2021) Improved binary pigeon-inspired optimization and its application for feature selection. Appl Intell, pp 1–19
50. Parpinelli RS, Lopes HS, Freitas AA (2002) Data mining with an ant colony optimization algorithm. IEEE Trans Evol Comput 6(4):321–332
51. Qin AK, Huang VL, Suganthan PN (2009) Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans Evol Comput 13(2):398–417
52. Qiu H, Duan H (2020) A multi-objective pigeon-inspired optimization approach to uav distributed flocking among obstacles. Inf Sci 509:515–529
53. Rao RV, Pawar PJ (2010) Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms. Appl Soft Comput 10:445–456
54. Reynolds AM, Bartumeus F (2009) Optimising the success of random destructive searches: Lévy walks can outperform ballistic motions. J Theor Biol 260(1):98–103
55. Storn R, Price K (1997) Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359
56. an S, Khamron A (2018) Omfo a new opposition-based moth-flame optimization algorithm for solving unconstrained optimization problems
57. Sharma N, Sharma H, Sharma A (2020) An effective solution for large scale single machine total weighted tardiness problem using lunar cycle inspired artificial bee colony algorithm. IEEE/ACM Trans Comput Biol Bioinform 17(5):1573–1581
58. Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In: 1998 IEEE International Conference on evolutionary computation proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), pp 69–73
59. Shi Y, Eberhart RC (1998) Parameter selection in particle swarm optimization. In: Evolutionary Programming
60. Tanabe R, Fukunaga AS (2014) Improving the search performance of shade using linear population size reduction. In: 2014 IEEE Congress on Evolutionary Computation (CEC) pp 1658–1665
61. Tirkolaee EB, Alireza G, et al (2020a) Fuzzy mathematical programming and self-adaptive artificial fish swarm algorithm for just-in-time energy-aware flow shop scheduling problem with outsourcing option. IEEE Trans Fuzzy Sys 28(11):2772–2783
62. Tirkolaee EB, Mardani A, Dashtian Z et al (2020b) A novel hybrid method using fuzzy decision making and multi-objective programming for sustainable-reliable supplier selection in two-echelon supply chain design. J Clean Prod 250:119–517
63. Viswanathan GM, Buldyrev SV, Havlin S et al (1999) Optimizing the success of random searches. Nature 401:911–914
64. Wang B, Wang DB, Ali ZA (2020a) A cauchy mutant pigeon-inspired optimization–based multi-unmanned aerial vehicle path planning method. Measurement and Control 53:83–92
65. Wang S, Liu G, Gao M et al (2020b) Heterogeneous comprehensive learning and dynamic multi-swarm particle swarm optimizer with two mutation operators. Inf Sci 540:175–201
66. Darrell W (1994) A genetic algorithm tutorial. Stat Comput 4(2):65–85
67. Wu WM, Li Z, Lin PN et al (2018) Moth-flame optimization algorithm based on chaotic crisscross operator. Comput Eng Appl 54(3):136–141
68. Xu X, Deng Y (2018) Uav power component—dc brushless motor design with merging adjacent-disturbances and integrated-dispatching pigeon-inspired optimization. IEEE Trans Magn 54:1–7
69. Yang Q, Chen W, Gu T et al (2017) Segment-based predominant learning swarm optimizer for large-scale optimization. IEEE Trans Cybern 47(9):2896–2910
70. Sun YB, Duan H, Xian N (2018) Fractional-order controllers optimized via heterogeneous comprehensive learning pigeon-inspired optimization for autonomous aerial refueling hose-drogue system. Aerosp Sci Technol 81:1–13
71. Ye W, Feng W, Fan S (2017) A novel multi-swarm particle swarm optimization with dynamic learning strategy. Applied Soft Computing 61:832–843
72. Yu K, Liang JJ, Qu B et al (2017) Parameters identification of photovoltaic models using an improved jaya optimization algorithm. Energy Convers Manag 150:742–753
73. Zhang B, Duan H (2017) Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment. IEEE/ACM Trans Comput Biol Bioinform 14(1):97–107
74. Zhang B, Zheng Y, Zhang M et al (2017) Fireworks algorithm with enhanced fireworks interaction. IEEE/ACM Trans Comput Biol Bioinform 14(1):42–55
75. Zhang D, Duan H (2018) Social-class pigeon-inspired optimization and time stamp segmentation for multi-uav cooperative path planning. Neurocomputing 313:229–246
76. Zhang H, Wang Z, Chen W et al (2021) Ensemble mutation-driven salp swarm algorithm with restart mechanism: Framework and fundamental analysis. Expert Syst Appl 165:113–897
77. Zhao Z, Zhang M, Zhang Z et al (2021) Hierarchical pigeon-inspired optimization-based mppt method for photovoltaic systems under complex partial shading conditions. IEEE Trans Ind Electron
78. Zheng S, Li J, Janecek A et al (2017) A cooperative framework for fireworks algorithm. IEEE/ACM Trans Comput Biol Bioinform 14(1):27–41
79. Zhong Y, Wang L, Lin M et al (2019) Discrete pigeon-inspired optimization algorithm with metropolis acceptance criterion for large-scale traveling salesman problem. Swarm Evol Comput 48:134–144
80. Zhou X, Lu J, Huang J et al (2021) Enhancing artificial bee colony algorithm with multi-elite guidance. Inf Sci 543:242–258
81. Cao ZJ, Hei XH, Wang L et al (2015) An improved brain storm optimization with differential evolution strategy for applications of ANNs. Math Probl Eng 2015

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Hanming Wang He received the B. S. degree from Nanchang University. He is currently working towards his M. Sc. degree in electrical engineering from the Naval University of Engineering, Wuhan, China. His current research interests include particle swarm optimization, genetic algorithms, and other computational intelligence techniques.

Jinghong Zhao He received the Ph.D. degree in electrical engineering from the Naval University of Engineering, Wuhan, China, in 2011. He is currently a professor of the electricity engineering school in the Naval University of Engineering. He has authored or coauthored over 50 high-level papers. His current research interests include artificial intelligence in power electronics, electric machine design.