The Journal of Supercomputing (2025)81:721
https://doi.org/10.1007/s11227-024-06859-3

Competitive differential evolution with knowledge inheritance for single-objective human-powered aircraft design

Rui Zhong[1] · Yang Cao[2] · Enzhi Zhang[2] · Masaharu Munetomo[1]

Accepted: 19 December 2024

Abstract
This paper introduces a novel and efficient optimizer derived from differential evolution (DE): Competitive differential evolution with knowledge inheritance (CDEKI). CDEKI is developed by introducing the competitive mechanism and incorporates a novel DE/winner-to-best/1 mutation strategy and a hybrid local search operation. Moreover, we emphasize the significance of a commonly neglected component in DE: the repair operation. We propose a novel repair operation with knowledge inheritance to accelerate optimization convergence, particularly when the generated offspring exceeds the search domain. Through comprehensive numerical experiments conducted on the CEC2020 benchmark functions, competing against sixteen state-of-the-art optimizers and advanced DE variants, our proposed CDEKI demonstrates significant superiority. Additionally, the ablation experiments are conducted to independently investigate the performance of the proposed strategies. Furthermore, we extend the application of CDEKI to real-world human-powered aircraft design tasks, showcasing its extraordinary performance in practical scenarios. As an effective and efficient optimizer, CDEKI presents a compelling alternative evolutionary approach for addressing real-world applications across diverse domains. The source code of CDEKI is available in https://github.com/RuiZhong961230/CDEKI.

Keywords Competitive differential evolution (CDE) · Novel mutation operation · Hybrid local search operation · Knowledge inheritance (KI) · Single-objective optimization · Human-powered aircraft (HPA) design

Extended author information available on the last page of the article

Published online: 10 April 2025

1 Introduction

Differential evolution (DE) [1] stands as a powerful optimization algorithm that belongs to the family of the evolutionary algorithm (EA). Being different from traditional mathematical optimization techniques that rely on the gradient, DE is a stochastic optimization methodology motivated by the principles of natural selection and evolution [2, 3]. Through the iteration of crossover, mutation, and selection operations, DE refines the population of candidate solutions to the specific problem and guides the population toward increasingly optimal solutions [4].

Thanks to its simplicity, robustness, and efficiency in solving complex optimization problems across various domains, DE is particularly well-suited for problems where the objective function is nonlinear, non-convex, noisy, or lacks derivative information [5]. These remarkable characteristics make DE a versatile tool for tackling real-world optimization challenges encountered in engineering [6–8], finance [9, 10], machine learning [11, 12], and other fields [13, 14]. Therefore, DE has gained widespread attention from researchers and scholars. In the meantime, many variants of DE have been proposed to tackle diverse optimization tasks, and we will briefly review the advancement of DE in Sect. 2.2.

This paper concentrates on the design of a single-objective human-powered aircraft (HPA) as detailed in [15], and proposes a novel competitive differential evolution with knowledge inheritance (CDEKI) approach. Specifically, we incorporate a competitive mechanism into the DE algorithm and introduce a DE/winner-to-best/1 mutation scheme to facilitate efficient global exploration. Additionally, the crossover operation is designed to enhance DE's exploitative performance. Therefore, we fully utilize the knowledge from both the current and the elite individuals to construct the offspring individual via the local search operation. Furthermore, we address a crucial yet often neglected component of DE: the repair operation. It is common for the constructed offspring individual to fall outside the search domain. Most DE algorithms randomly re-generate or clip values in specific dimensions when exceeding the search space. However, such inefficient operations fail to adequately leverage domain knowledge and past optimization experiences to generate high-quality solutions. To address this limitation, we propose an effective repair operation based on knowledge inheritance and integrate it into our proposed CDEKI. Specifically, the main contributions of this paper can be summarized as follows:

- We propose a novel competitive differential evolution with knowledge inheritance (CDEKI).
- The DE/winner-to-best/1 mutation scheme, the hybrid local search operation, and the repair operation based on knowledge inheritance are integrated into CDEKI.
- We conduct comprehensive numerical experiments on CEC2020 benchmark functions to evaluate the performance of CDEKI competing with state-of-the-art optimizers.
- We conduct ablation experiments to independently investigate the performance of the proposed strategies.

- We extend CDEKI to solve real-world single-objective HPA design and achieve encouraging performance.

The remainder of this paper is organized as follows: Sect. 2 introduces the related works including the basic DE and the mathematical model of the single-objective human-powered aircraft design. Section 3 introduces our proposed CDEKI in detail. Section 4 presents numerical experiments and statistical analyses, and the performance analyses are discussed in Sect. 5. Finally, Sect. 6 concludes this paper.

2 Related works

2.1 Basic DE

We begin the introduction of the basic DE by the definition of optimization problems. Without loss of generality, the minimization problem is mathematically defined by Eq. (1).

$$f(\boldsymbol{x}^*) = \min f(\boldsymbol{x}), s.t. \boldsymbol{x} \in \mathcal{R}^D \tag{1}$$

where $\boldsymbol{x} = \{x_1, x_2, \ldots, x_D\}$ is a solution vector with D dimensions. Optimization algorithms aim to find optimum $\boldsymbol{x}^*$ with a limited computational budget.

Subsequently, we outline the four primary components of DE: initialization, mutation, crossover, and selection. It is important to note that all explanations are presented within the context of the minimization.

Initialization The first step of DE implementation is population initialization, which is described in Eq. (2).

$$X = \begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{x_2} \\ \boldsymbol{x_3} \\ \vdots \\ \boldsymbol{x_N} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1D} \\ x_{21} & x_{22} & \cdots & x_{2D} \\ x_{31} & x_{32} & \cdots & x_{3D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{ND} \end{bmatrix}, x_{ij} \tag{2}$$

where $\boldsymbol{x}_i$ denotes the i^{th} individual and x_{ij} represents the value in the j^{th} dimension of the $\boldsymbol{x}_i$. $\boldsymbol{lb}_j$ and $\boldsymbol{ub}_j$ are the lower and the upper bound of the j^{th} dimension, respectively, and r is a random number in (0, 1).

Mutation: When DE enters the main loop, the mutation operation is first activated to construct the mutated vector, and the representative mutation schemes are listed in Eq. (3).

$$
\begin{aligned}
&\text{DE/rand/1}: \boldsymbol{v}_i^g = \boldsymbol{x}_{r1}^g + F \cdot (\boldsymbol{x}_{r2}^g - \boldsymbol{x}_{r3}^g) \\
&\text{DE/cur/1}: \boldsymbol{v}_i^g = \boldsymbol{x}_i^g + F \cdot (\boldsymbol{x}_{r1}^g - \boldsymbol{x}_{r2}^g) \\
&\text{DE/best/1}: \boldsymbol{v}_i^g = \boldsymbol{x}_{best}^g + F \cdot (\boldsymbol{x}_{r1}^g - \boldsymbol{x}_{r2}^g) \\
&\text{DE/cur-to-best/1}: \boldsymbol{v}_i^g = \boldsymbol{x}_i^g + F \cdot (\boldsymbol{x}_{best}^g - \boldsymbol{x}_i^g) + F \cdot (\boldsymbol{x}_{r1}^g - \boldsymbol{x}_{r2}^g) \\
&\text{DE/rand-to-best/1}: \boldsymbol{v}_i^g = \boldsymbol{x}_{r1}^g + F \cdot (\boldsymbol{x}_{best}^g - \boldsymbol{x}_{r1}^g) + F \cdot (\boldsymbol{x}_{r2}^g - \boldsymbol{x}_{r3}^g)
\end{aligned} \tag{3}
$$

where $\boldsymbol{x}_{r1}^g$, $\boldsymbol{x}_{r2}^g$, and $\boldsymbol{x}_{r3}^g$ are randomly selected individuals from the population and mutually distinct in the g^{th} iteration, $\boldsymbol{x}_{best}^g$ denotes the best solution found so far, and F is the scaling factor to control the amplification of differential vector.

Crossover Although many novel crossover strategies such as exponential crossover [16] and blending crossover [17] have been proposed, the most commonly utilized binomial crossover is expressed in Eq. (4).

$$
u_{i,j}^g = \begin{cases} v_{i,j}^g, & \textit{if } r \leq Cr \textit{ or } j = j_{\text{rand}} \\ x_{i,j}^g, & \text{otherwise} \end{cases} \tag{4}
$$

Cr represents the crossover rate to control the probability of inherited genes between the mutated vector $\boldsymbol{u}_{i,j}^g$ and the parent individual $\boldsymbol{x}_{i,j}^g$. j_{rand} is a random integer in $\{1, 2, \ldots, D\}$.

Selection The selection mechanism in basic DE ensures the survival of elite individuals to the next iteration, as formulated in Eq. (5).

$$
\boldsymbol{x}_{i,j}^{g+1} = \begin{cases} \boldsymbol{u}_{i,j}^g, \ \textit{if } f(\boldsymbol{u}_{i,j}^g) \leq f(\boldsymbol{x}_{i,j}^g) \\ \quad \boldsymbol{x}_{i,j}^g, \ \textit{otherwise} \end{cases} \tag{5}
$$

The one-to-one greedy selection mechanism in DE can survive the elites while maintaining the population diversity. In summary, the pseudocode of the basic DE is presented in Algorithm 1.

Algorithm 1 Basic DE

Require: Population size: N, Dimension: D, Maximum iteration: G
Ensure: Optimum: $\boldsymbol{x}_{best}^{g}$

```
X ← initial(N, D) # Population initialization
g = 0
x^g_best ← best(X)
while g < G do
    for i = 0 to N do
        Construct the mutated vector using Eq. (3)
        Crossover using Eq. (4)
        Selection using Eq. (5)
    end for
    g ← g + 1
    x^g_best ← best(X)
end while
return x^g_best
```

2.2 Single-objective human-powered aircraft design

Similar to a bicycle, the human-powered aircraft (HPA) achieves propulsion through the pilot pedaling to rotate a propeller. These aircraft typically boast a wingspan ranging from 20 to 35 ms, comparable to that of conventional passenger aircraft, yet their mass is relatively light, ranging from 25 to 35 kgs. The wings of the HPA are commonly constructed from foam and balsa wood, covered with heat-shrinkable film, and supported by frameworks made of carbon fiber-reinforced plastic (CFRP) pipes.

A set of variables defines the main wing shape, which divides the right wing into n segments, delineated by $(n+1)$ cross sections. Figure 1 shows an instance when $n = 4$. These real-coded variables contain:

- a_i: the airfoil selection.
- b_i: the segment length.
- c_i: the chord length.
- α_i: the angle of attack.
- d_i: the diameter of the CFRP pipe forming the main spar.

Equation (6) defines the optimization problems in the HPA benchmark.

$$
\begin{aligned}
\text{HPA101}: & f(\boldsymbol{x}) = D(\boldsymbol{x}) \\
\text{HPA102}: & f(\boldsymbol{x}) = P(\boldsymbol{x}) = D(\boldsymbol{x})V(\boldsymbol{x})/\eta \\
\text{HPA103}: & f(\boldsymbol{x}) = -V(\boldsymbol{x}) \\
\text{HPA131}: & f(\boldsymbol{x}) = D(\boldsymbol{x}) \\
& g_1(\boldsymbol{x}) = n_m n_s \epsilon_{\max}(\boldsymbol{x})/\epsilon_u - 1 \\
& g_2(\boldsymbol{x}) = B(\sin\gamma(\boldsymbol{x}) - \sin\gamma_u)/2 \\
& g_3(\boldsymbol{x}) = -\delta_{park}(\boldsymbol{x}) \\
\text{HPA142}: & f(\boldsymbol{x}) = P(\boldsymbol{x}) = D(\boldsymbol{x})V(\boldsymbol{x})/\eta \\
& g_1(\boldsymbol{x}) = n_m n_s \epsilon_{\max}(\boldsymbol{x})/\epsilon_u - 1 \\
& g_2(\boldsymbol{x}) = 1 - (V(\boldsymbol{x})/V_{\min})^3 \\
\text{HPA143}: & f(\boldsymbol{x}) = -V(\boldsymbol{x}) \\
& g_1(\boldsymbol{x}) = n_m n_s \epsilon_{\max}(\boldsymbol{x})/\epsilon_u - 1 \\
& g_2(\boldsymbol{x}) = B(\sin\gamma(\boldsymbol{x}) - \sin\gamma_u)/2 \\
& g_3(\boldsymbol{x}) = -\delta_{park}(\boldsymbol{x}) \\
& g_4(\boldsymbol{x}) = P(\boldsymbol{x}) - P_{\max}
\end{aligned} \tag{6}
$$

The tasks of HPA101, HPA102, and HPA103 are the unconstrained versions of HPA131, HPA142, and HPA143. In this context, the functions $P(\cdot)$, $D(\cdot)$, $V(\cdot)$, $E(\cdot)$, and $B(\cdot)$ represent the required power, drag, cruise speed, wing efficiency, and wing span, respectively. The decision variables δ and δ_{park} indicate the wingtip displacement (positive upwards) during flight and parked conditions, respectively, relative to the height of the wing root. $\epsilon_{\max}$ denotes the maximum strain in CFRP pipes under both parked and flight conditions; while, γ represents the dihedral angle at the wingtip during the flight condition. Moreover, the constants in the mathematical model definition are summarized as follows:

- $\eta = 0.85 \times 0.95$: the product of propeller and drivetrain efficiency.
- $n_m = 1.5$, $n_s = 2$, and $\epsilon_u = 0.0027$: the load factor, the safety factor, and the reference strain.
- $\gamma_u = 8$: the maximum dihedral angle.
- $P_{\max} = 400$: the maximum required power.
- $V_{\min} = 7.3$: the minimum cruise speed.

Additionally, the HPA benchmark provided in https://github.com/Nobuo-Namura/hpa contains different difficulty levels, and interested readers can refer to [15] for more details about the HPA benchmark.

3 Competitive differential evolution with knowledge inheritance (CDEKI)

This section provides a detailed introduction to the proposed CDEKI. Drawing inspiration from the simple yet effective architecture of DE, the main flowchart of CDEKI is depicted in Fig. 2. In the following sections, we introduce our proposed competitive mutation strategy, the hybrid local search operations, and a repair mechanism enhanced by knowledge inheritance.

3.1 Competitive mutation strategy

We introduce the competitive mechanism to the mutation scheme and propose a novel DE/winner-to-best/1 operation, as formulated in Eq. (7).

$$\boldsymbol{v}_i^g = \begin{cases} \boldsymbol{x}_{r1}^g + F_1 \cdot (\boldsymbol{x}_{best}^g - \boldsymbol{x}_{r1}^g) + F_2 \cdot (\boldsymbol{x}_{r2}^g - \boldsymbol{x}_{r3}^g), & \textit{if } f(\boldsymbol{x}_{r1}^g) \le f(\boldsymbol{x}_i^g) \\ \boldsymbol{x}_i^g + F_1 \cdot (\boldsymbol{x}_{best}^g - \boldsymbol{x}_i^g) + F_2 \cdot (\boldsymbol{x}_{r2}^g - \boldsymbol{x}_{r3}^g), & \textit{otherwise} \end{cases} \tag{7}$$

where F_1 and F_2 are two random values following a normal distribution $N(0.5, 0.3)$ as recommended in [18]. Simply, the proposed DE/winner-to-best/1 strategy randomly selects a competitor individual $\boldsymbol{x}_{r1}^g$. If it has a better fitness value, it will replace the current individual $\boldsymbol{x}_i^g$ and act as the base vector to construct the mutated vector $\boldsymbol{v}_i^g$ using the DE/rand-to-best/1 scheme; otherwise, the current individual $\boldsymbol{x}_i^g$ will survive and the DE/cur-to-best/1 mutation scheme is activated to construct the mutated vector $\boldsymbol{v}_i^g$.

The structure of the proposed DE/winner-to-best/1 strategy resembles a fusion of the DE/cur-to-best/1 and DE/rand-to-best/1 strategies. However, incorporating a competitive mechanism enables the automatic selection of the most suitable mutation strategy, ensuring the utilization of superior knowledge to construct the mutated vector. Furthermore, CDEKI can benefit from the proposed DE/winner-to-best/1 mutation strategy from (i). Strengthened convergence: By leveraging a superior base vector, the optimization process experiences rapid convergence and contributes to accelerating the approach to optimal solutions. (ii). The prevention of premature optimization: The inclusion of the random differential vector $F_2 \cdot (\boldsymbol{x}_{r2}^g - \boldsymbol{x}_{r3}^g)$ serves to mitigate premature optimization and promote the exploration of diverse search spaces. (iii). Versatile scaling factor: The utilization of a simple yet effective random generator-based scaling factor F_i facilitates the generation of differential vectors with varying scales, thereby enhancing the adaptability and robustness of the mutation strategy.

3.2 Hybrid local search operation

The crossover operation aims to strike a balance between exploration and exploitation. Therefore, we substitute the conventional binomial crossover in DE with the hybrid local search operation, as defined in Eq. (8).

$$\boldsymbol{u}_{i,j}^{g}=\begin{cases}\boldsymbol{v}_{i,j}^{g}+r\cdot\boldsymbol{\delta}_{j}, & \textit{if } \textbf{rand}()\leq Cr\\ \boldsymbol{x}_{best,j}^{g}+r\cdot\boldsymbol{\delta}_{j}, & \textit{otherwise}\end{cases}\tag{8}$$

Here, Cr denotes the crossover rate, which follows a normal distribution $N(0.5, 0.3)$. $\boldsymbol{\delta}$ is the differential vector constructed by the current i^{th} individual $\boldsymbol{x}_i^g$ and selected competitor individual $\boldsymbol{x}_{r1}^g$ using Eq. (9).

$$\boldsymbol{\delta}=\begin{cases}\boldsymbol{x}_{i}^{g}-\boldsymbol{x}_{r1}^{g}, & \textit{if } f(\boldsymbol{x}_{i}^{g})\leq f(\boldsymbol{x}_{r1}^{g})\\ \boldsymbol{x}_{r1}^{g}-\boldsymbol{x}_{i}^{g}, & \textit{otherwise}\end{cases}\tag{9}$$

$\boldsymbol{\delta}$ denotes the differential vector that starts from the worse solution to the better one. If we ignore the component of $r\cdot\boldsymbol{\delta}_j$ in Eq. (8), this structure follows the binomial crossover between the mutated vector $\boldsymbol{v}_{best,j}^g$ and the current best individual $\boldsymbol{v}_{i,j}^g$. Considering that frequent knowledge inheritance from the current best individual $\boldsymbol{v}_{best,j}^g$ may inadvertently guide the optimization toward local optima and cause premature convergence. To augment the optimization, we introduce a positive differential vector to construct the offspring individual, which corresponds to the perturbation within the local search operation. This innovative hybrid local search operation not only maintains population diversity throughout the optimization process but also significantly expedites convergence. By iteratively integrating the superior knowledge derived from elite individuals into the construction of offspring individuals, this method displays considerable potential in improving the performance of our proposed CDEKI.

3.3 Repair operation with knowledge inheritance

Most EAs adopt the repair operations described in Eqs. (10) and (11) when the value of the specific dimension is outside the search domain.

$$\begin{aligned}\text{Random re-initialization}:&\textit{if } \boldsymbol{x}_{i,j}^{g}>\boldsymbol{ub}_j \textit{ or } \boldsymbol{x}_{i,j}^{g}<\boldsymbol{lb}_j\\ &\textit{then } \boldsymbol{x}_{i,j}^{g}=r\cdot(\boldsymbol{ub}_j-\boldsymbol{lb}_j)+\boldsymbol{lb}_j\end{aligned}\tag{10}$$

$$\begin{aligned}\text{Clipping}:&\textit{if } \boldsymbol{x}_{i,j}^{g}>\boldsymbol{ub}_j \textit{ then } \boldsymbol{x}_{i,j}^{g}=\boldsymbol{ub}_j\\ &\textit{if } \boldsymbol{x}_{i,j}^{g}<\boldsymbol{lb}_j \textit{ then } \boldsymbol{x}_{i,j}^{g}=\boldsymbol{lb}_j\end{aligned}\tag{11}$$

While knowledge-free techniques in optimization algorithms can be efficient, they often risk compromising the superior genetic information present in the original solutions. Recognizing this challenge, we introduce an effective repair operation with knowledge inheritance. In this operation, when a certain gene exceeds the predefined search domain, there is a probability of inheriting the superior gene from the current best solution $\boldsymbol{x}_{best}^g$. This ensures that valuable genetic information is preserved and utilized optimally. For a detailed implementation, Algorithm 2 outlines the pseudocode of our proposed repair operation.

Algorithm 2 Repair operation with knowledge Inheritance

Require: Individual $\boldsymbol{x}_i^g$, Dimension: D, Lower and upper bound: $\boldsymbol{lb}$ and $\boldsymbol{ub}$, Current best individual: $\boldsymbol{x}_{best}^g$
Ensure: Repaired individual $\boldsymbol{x}_i^g$
for $j = 0$ *to* D **do**
  if $\boldsymbol{x}_{i,j}^g > \boldsymbol{ub}_j$ or $\boldsymbol{x}_{i,j}^g < \boldsymbol{lb}_j$ **then**
    if rand() < 0.5 **then**
      $\boldsymbol{x}_{i,j}^g = r \cdot (\boldsymbol{ub}_j - \boldsymbol{lb}_j) + \boldsymbol{lb}_j$ # Eq. (10)
    else
      $\boldsymbol{x}_{i,j}^g = \boldsymbol{x}_{best,j}^g$ # Knowledge inheritance
    end if
  end if
end for
return $\boldsymbol{x}_i^g$

In summary, the pseudocode of the proposed CDEKI is presented in Algorithm 3.

Algorithm 3 CDEKI

Require: Population size: N, Dimension: D, Maximum iteration: G
Ensure: Optimum: $\boldsymbol{x}_{best}^g$
Initialize the population X randomly
$g \leftarrow 0$
$\boldsymbol{x}_{best}^g \leftarrow$ **best**(X)
while $g < G$ **do**
  for $i = 0$ *to* N **do**
    Select a random competitor $\boldsymbol{x}_{r1}^g$
    Construct the mutated vector $\boldsymbol{v}_i^g$ using Eq. (7)
    Construct the offspring individual $\boldsymbol{u}_i^g$ using Eq. (8)
    Repair the offspring individual $\boldsymbol{u}_i^g$ using Algorithm (2)
    Greedy selection using Eq. (5)
  end for
  $\boldsymbol{x}_{best}^g \leftarrow$ **best**(X)
  $g \leftarrow g + 1$
end while
return $\boldsymbol{x}_{best}^g$

---

4 Numerical experiments

To evaluate the performance of our proposed CDEKI, we conduct comprehensive numerical experiments across a range of benchmark functions and apply CDEKI to solve real-world HPA design problems. In the following sections, we provide detailed insights into the experimental setup and outcomes. Section 4.1 outlines the experimental settings, including details on the experimental environments and implementation, the benchmark functions, and competitor algorithms and parameters. Section 4.2 presents the experimental results and statistical analyses, demonstrating the competitive efficacy of our proposed CDEKI.

4.1 Experimental settings

4.1.1 Experimental environments and implementation

We conducted numerical experiments using Python 3.11 on a Lenovo Legion R9000P laptop running Windows 11. The laptop features an AMD Ryzen 7 5800 H processor with Radeon Graphics, operating at a clock speed of 3.20 GHz, and is equipped with 16GB of RAM. Providing this detailed information about the experimental setup and implementation ensures the reproducibility of our research

4.1.2 Benchmark functions

We adopted two well-known benchmarks and one real-world HPA suite to investigate the performance of CDEKI comprehensively, which are summarized as follows:

- 30-D and 50-D CEC2020 benchmark functions [19].
- HPA suite with variable level in $\{3, 4, 5\}$ and difficulty level in $\{0, 1\}$.

4.1.3 Competitor algorithms and parameters

To comprehensively and fairly investigate the performance of our proposed CDEKI, we carefully select four categories of EAs as competitor algorithms, which are listed as follows:

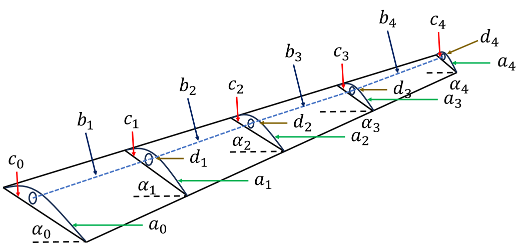
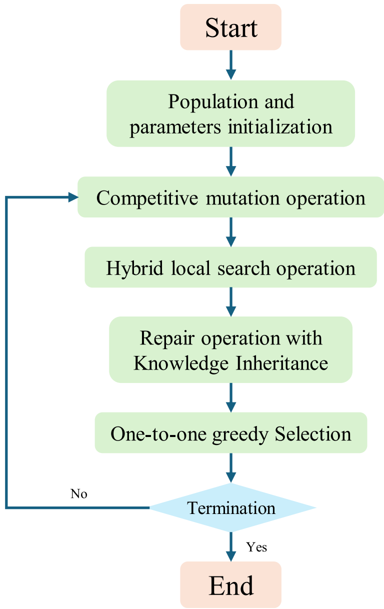


Fig. 1 Variables in defining the main wing shape

Fig. 2 Flowchart of CDEKI



- Classic EAs: genetic algorithm (GA) [20], particle swarm optimization (PSO) [21], DE [1], and covariance matrix adaptation evolution strategy (CMA-ES) [22].
- Well-known EAs: grey wolf optimizer (GWO) [23], whale optimization algorithm (WOA) [24], golden jackal optimization (GJO) [25], and RIME algorithm [26].
- Advanced variants of DE: SHADE [27], iL-SHADE [28], Hip-DE [29], and gene targeting DE (GTDE) [30].
- Advanced variants of PSO: chaotic PSO (CPSO) [31], comprehensive learning PSO (CLPSO) [32], CLPSO with local search (CLPSO-LS) [33], and phasor PSO (PPSO) [34].

Additionally, the specific parameter settings of competitor algorithms are summarized in Table 1.All competitor algorithms adhere to the recommended parameter settings outlined in their respective papers. For CEC2020 benchmark functions, except for iL-SHADE, both the competitor algorithms and CDEKI maintain a consistent population size of 100, while the maximum fitness evaluations (FEs) for all algorithms are fixed at $1000 \times D$. In the HPA design problems, the population size and maximum FEs for all optimizers are standardized at 20 and 1000, as specified in [15]. To ensure robustness, each optimization task is independently executed 30 times in CEC2020 benchmark functions and 20 times in HPA design tasks to mitigate the impact of randomness.

Table 1 Parameters of competitor algorithms

EAs	Parameters	Value
CDEKI	μ_F and σ_F	0.5 and 0.3
	μ_{Cr} and σ_{Cr}	0.5 and 0.3
GA	Crossover probability *pc*	0.9
	Mutation probability *pm*	0.01
	Selection	Tournament
PSO	Inertia factor *w*	1
	Coefficients c_1 and c_2	2.05
	Max. and Min. speed	2 and -2
DE	Mutation strategy	DE/cur-to-rand/1/bin
	Scaling factor *F*	0.8
	Crossover rate *Cr*	0.9
CMA-ES	σ	1.3
GWO	Parameter-free	
WOA	Constant *b*	1
GJO	Parameter-free	
RIME	Parameter *w*	5
CPSO	Inertia factor *w*	0.5+**rand**()/2
	Coefficients c_1 and c_2	2
CLPSO	Local coefficient c_{local}	1.2
	Max. and min. Weight	0.9 and 0.4
CLPSO-LS	Local coefficient c_{local}	1.2
	Max. and min. weight	0.9 and 0.4
	Local search scheme	BFGS method
PPSO	Parameter-free	
SHADE	Mutation strategy	DE/cur-to-*p*best/1/bin
	μ_F and μ_{Cr}	0.5 and 0.5
iL-SHADE	Initial population size	$12 \times D$
	Mutation strategy	DE/cur-to-*p*best/1/bin
	μ_F and μ_{Cr}	0.5 and 0.8
Hip-DE	μ_F and σ_F	0.6 and 0.1
	μ_{Cr} and σ_{Cr}	0.8 and 0.1
	F_{best} and Cr_{best}	0.5 and 0.9
GTDE	μ_F and σ_F	0.7 and 0.5
	μ_{Cr} and σ_{Cr}	0.5 and 0.5

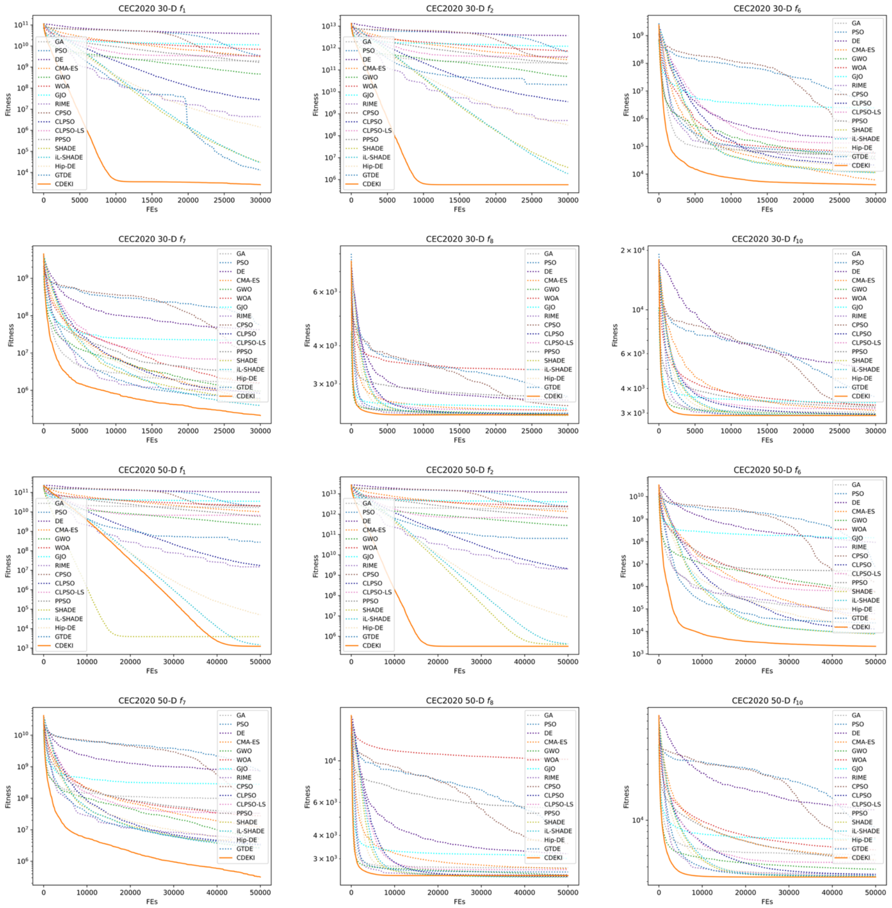


Fig. 3 Convergence curves of competitor algorithms in CEC2020 representative functions (i.e., f_1: Unimodal function; f_2: Multimodal function; f_6 and f_7: Hybrid functions; f_8 and f_{10}: Composite functions)

4.2 Experimental results and statistical analyses

This section presents the experimental results and statistical analyses. We commence by showcasing the outcomes of comparison experiments conducted on both the CEC2020 benchmark functions, as well as HPA design tasks. To determine the significance between our proposed CDEKI and other competitor algorithms, we employ the Mann–Whitney U test for each pair of algorithms. Subsequently, the Holm multiple comparison test is applied to correct the p-values derived from the Mann–Whitney U test. The symbols $+$, $\approx$, and—indicate that CDEKI performs significantly better, shows no significant difference, or performs significantly worse compared to the compared algorithm, respectively. Additionally, the average rank metric is computed, with the best-performing fitness value denoted in bold.

Performance on CEC2020 benchmark functions Tables 2 and 3 present the experimental results and statistical analyses on CEC2020 benchmark functions;

Table 2 Experimental results and statistical analyses on 30-D CEC2020 benchmark functions

Func.		GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
f_1	Mean	1.983e+09 +	3.479e+09 +	3.796e+10 +	3.154e+09 +	4.704e+08 +	7.037e+09 +	1.134e+10 +	4.507e+06 +	3.380e+09 +
	Std	3.017e+08	1.505e+09	4.492e+09	5.271e+08	3.956e+08	4.018e+09	2.541e+09	1.777e+06	1.819e+09
f_2	Mean	2.087e+11 +	6.703e+11 +	3.706e+12 +	2.971e+11 +	5.126e+10 +	7.427e+11 +	1.217e+12 +	4.976e+08 +	3.737e+11 +
	Std	3.854e+10	4.442e+11	4.166e+11	4.725e+10	6.168e+10	4.105e+11	3.629e+11	3.341e+08	1.780e+11
f_3	Mean	6.981e+10 +	1.855e+11 +	1.269e+12 +	1.009e+11 +	1.765e+10 +	2.369e+11 +	4.357e+11 +	1.485e+08 +	9.968e+10 +
	Std	1.205e+10	1.106e+11	1.409e+11	1.864e+10	1.446e+10	1.289e+11	1.074e+11	7.087e+07	4.119e+10
f_4	Mean	1.941e+03 +	8.403e+03 +	1.046e+05 +	2.104e+03 +	1.919e+03 +	7.069e+03 +	6.042e+03 +	**1.913e+03** −	2.711e+03 +
	Std	6.685e+00	1.238e+04	4.793e+04	1.120e+02	4.666e+00	6.069e+03	4.888e+03	2.231e+00	1.729e+03
f_5	Mean	8.487e+05 +	2.238e+07 +	1.426e+07 +	5.498e+05 +	5.930e+05 +	2.210e+06 +	3.406e+06 +	5.991e+05 +	3.167e+06 +
	Std	2.604e+05	2.113e+07	4.766e+06	1.614e+05	3.503e+05	4.339e+06	1.438e+06	3.293e+05	4.096e+06
f_6	Mean	4.540e+04 +	1.424e+06 +	1.907e+05 +	6.081e+03 +	3.680e+04 +	5.669e+04 +	2.476e+06 +	3.405e+04 +	5.897e+04 +
	Std	4.785e+04	4.678e+06	3.004e+05	1.186e+03	1.017e+04	7.296e+04	2.717e+06	1.544e+04	1.016e+05
f_7	Mean	1.509e+06 +	5.479e+07 +	4.159e+07 +	5.817e+05 +	9.624e+05 +	1.586e+06 +	2.196e+07 +	9.515e+05 +	4.256e+06 +
	Std	7.894e+05	5.888e+07	1.171e+07	1.319e+05	5.147e+05	1.186e+06	1.582e+07	5.809e+05	6.931e+06
f_8	Mean	2.403e+03 +	2.726e+03 +	2.652e+03 +	2.455e+03 +	2.381e+03 +	3.332e+03 +	2.493e+03 +	2.384e+03 +	2.547e+03 +
	Std	7.781e+00	1.867e+02	2.917e+01	9.179e+00	9.729e+00	5.418e+02	2.803e+01	8.445e+00	6.459e+01
f_9	Mean	6.172e+03 +	1.041e+04 +	1.316e+04 +	6.313e+03 +	3.262e+03 +	1.166e+04 +	1.144e+04 +	2.752e+03 +	7.619e+03 +
	Std	2.824e+02	3.823e+03	4.679e+02	3.331e+02	5.494e+02	5.863e+03	1.368e+03	3.932e+01	1.140e+03

Table 2 continued

Func.		GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
f_{10}	Mean	3.029e+03 +	3.627e+03 +	5.041e+03 +	3.100e+03 +	2.969e+03 +	3.303e+03 +	3.392e+03 +	2.927e+03 +	3.280e+03 +
	Std	1.885e+01	5.366e+02	2.991e+02	3.047e+01	3.408e+01	1.554e+02	1.781e+02	5.012e+00	9.034e+01
+/≈/−		10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	9/0/1	10/0/0
Ave. rank		9.4	15.4	16.3	8.7	7.5	14.2	15.1	5.8	12.5

Func.		CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
f_1	Mean	2.838e+07 +	3.450e+09 +	1.665e+09 +	3.031e+04 +	2.890e+04 +	1.400e+06 +	1.286e+04 +	**2.629e+03**
	Std	7.300e+06	7.477e+08	9.664e+08	1.990e+04	2.190e+04	6.720e+05	1.297e+04	3.214e+03
f_2	Mean	3.558e+09 +	4.118e+11 +	1.927e+11 +	3.573e+06 +	1.890e+06 +	3.025e+08 +	2.149e+10 +	**5.838e+05**
	Std	6.616e+08	7.687e+10	1.202e+11	3.030e+06	1.064e+06	1.769e+08	6.269e+10	6.163e+05
f_3	Mean	1.662e+09 +	1.491e+11 +	7.237e+10 +	1.479e+06 +	1.180e+06 +	1.331e+08 +	7.363e+09 +	**2.806e+05**
	Std	3.188e+08	2.207e+10	3.922e+10	1.071e+06	8.512e+05	6.527e+07	1.956e+10	2.966e+05
f_4	Mean	1.924e+03 +	2.404e+03 +	3.976e+03 +	1.915e+03 ≈	1.914e+03 ≈	1.917e+03 ≈	1.920e+03 ≈	1.917e+03
	Std	2.106e+00	2.207e+02	2.762e+03	1.254e+00	1.230e+00	1.578e+00	3.841e+00	1.786e+00
f_5	Mean	3.157e+05 +	1.352e+06 +	1.524e+06 +	2.682e+05 +	2.556e+05 +	3.247e+05 +	4.772e+05 +	**1.839e+05**
	Std	7.346e+04	3.929e+05	1.285e+06	6.244e+04	6.881e+04	8.040e+04	1.955e+05	8.722e+04
f_6	Mean	2.104e+04 +	1.306e+05 +	4.205e+04 +	1.157e+04 +	1.128e+04 +	1.735e+04 +	5.754e+04 +	**4.138e+03**
	Std	6.142e+03	7.705e+04	3.896e+04	4.446e+03	3.142e+03	6.352e+03	2.119e+05	3.721e+03
f_7	Mean	8.078e+05 +	6.509e+06 +	2.904e+06 +	5.305e+05 +	3.864e+05 +	8.913e+05 +	6.124e+05 +	**2.126e+05**
	Std	2.479e+05	2.617e+06	2.973e+06	2.073e+05	1.088e+05	2.771e+05	5.145e+05	1.277e+05
f_8	Mean	2.382e+03 +	2.470e+03 +	2.709e+03 +	**2.365e+03** −	2.366e+03 −	2.367e+03 −	2.406e+03 +	2.376e+03
	Std	3.607e+00	1.351e+01	1.380e+02	3.304e+00	3.441e+00	1.548e+00	1.627e+01	9.237e+00
f_9	Mean	3.025e+03 +	7.734e+03 +	6.993e+03 +	**2.602e+03** −	2.603e+03 −	2.623e+03 −	4.240e+03 +	2.643e+03
	Std	7.546e+01	4.590e+02	2.151e+03	4.323e−01	7.534e−01	8.974e+00	2.010e+03	1.107e+02

Table 2 continued

Func.		CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
f_{10}	Mean	2.964e+03 +	3.166e+03 +	3.219e+03 +	2.922e+03 +	2.922e+03 +	2.926e+03 +	2.934e+03 +	**2.922e+03**
	Std	1.091e+01	3.992e+01	1.303e+02	1.283e−01	7.053e−01	1.189e+00	3.677e+01	1.809e+00
+/≈/−		10/0/0	10/0/0	10/0/0	8/0/2	8/0/2	8/0/2	9/1/0	–
Ave. rank		6.2	12.4	11.2	2.7	2.3	4.4	6.9	**2.0**

f_1: Unimodal function; $f_2 - f_4$: Multimodal functions; $f_5 - f_7$: Hybrid functions; $f_8 - f_{10}$: Composite functions

Table 3 Experimental results and statistical analyses on 50-D CEC2020 benchmark functions

Func.		GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
f_1	Mean	1.690e+10 +	1.979e+10 +	1.017e+11 +	1.018e+10 +	2.250e+09 +	2.048e+10 +	3.505e+10 +	1.501e+07 +	2.043e+10 +
	Std	1.782e+09	4.876e+09	9.010e+09	2.459e+09	1.263e+09	8.090e+09	5.923e+09	5.375e+06	5.169e+09
*f_2	Mean	1.832e+12 +	2.482e+12 +	1.192e+13 +	1.339e+12 +	2.794e+11 +	2.364e+12 +	4.071e+12 +	2.056e+09 +	2.356e+12 +
	Std	1.748e+11	6.330e+11	1.004e+12	3.446e+11	1.469e+11	1.074e+12	4.510e+11	1.130e+09	6.131e+11
f_3	Mean	6.120e+11 +	7.088e+11 +	4.321e+12 +	4.346e+11 +	1.092e+11 +	6.332e+11 +	1.238e+12 +	5.210e+08 +	7.321e+11 +
	Std	5.464e+10	2.018e+11	3.090e+11	1.293e+11	5.545e+10	2.142e+11	1.545e+11	1.708e+08	1.392e+11
f_4	Mean	5.343e+03 +	7.472e+04 +	1.488e+06 +	7.536e+03 +	1.959e+03 +	3.380e+04 +	4.475e+04 +	1.931e+03 ≈	1.987e+04 +
	Std	1.222e+03	1.384e+05	6.471e+05	3.163e+03	4.349e+01	4.676e+04	3.119e+04	5.593e+00	1.818e+04
f_5	Mean	1.416e+07 +	7.795e+07 +	5.450e+07 +	2.120e+06 +	4.214e+06 +	1.026e+07 +	2.960e+07 +	4.395e+06 +	1.485e+07 +
	Std	3.986e+06	5.727e+07	1.709e+07	3.948e+05	3.037e+06	6.716e+06	1.641e+07	1.880e+06	1.374e+07
f_6	Mean	8.729e+04 +	6.528e+06 +	9.271e+07 +	3.477e+04 +	5.629e+05 +	6.003e+05 +	1.481e+08 +	6.292e+04 +	1.601e+06 +
	Std	5.375e+04	9.114e+06	2.877e+07	9.391e+03	8.440e+05	1.075e+06	1.876e+08	2.799e+04	3.439e+06
f_7	Mean	9.355e+07 +	7.203e+08 +	7.549e+08 +	1.211e+07 +	5.076e+06 +	2.767e+07 +	2.771e+08 +	5.645e+06 +	6.968e+07 +
	Std	2.224e+07	6.680e+08	2.112e+08	3.095e+06	2.914e+06	2.628e+07	1.215e+08	3.074e+06	5.800e+07
f_8	Mean	2.682e+03 +	4.142e+03 +	3.192e+03 +	2.659e+03 +	2.453e+03 +	1.019e+04 +	3.007e+03 +	2.466e+03 +	3.619e+03 +
	Std	2.058e+01	5.913e+02	8.934e+01	4.332e+01	2.033e+01	2.367e+03	1.451e+02	1.702e+01	4.574e+02
f_9	Mean	1.544e+04 +	2.273e+04 +	2.699e+04 +	1.141e+04 +	6.200e+03 +	2.866e+04 +	2.613e+04 +	2.934e+03 +	2.049e+04 +
	Std	5.469e+02	3.618e+03	2.014e+03	1.403e+03	1.225e+03	9.777e+03	4.090e+03	8.038e+01	2.955e+03

Table 3 continued

Func.		GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
f_{10}	Mean	5.064e+03 +	8.750e+03 +	1.225e+04 +	4.601e+03 +	3.828e+03 +	5.592e+03 +	6.896e+03 +	3.462e+03 +	7.076e+03 +
	Std	2.193e+02	2.612e+03	1.641e+03	3.072e+02	2.097e+02	9.995e+02	7.367e+02	1.579e+02	9.198e+02
+/≈/−		10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	9/1/0	10/0/0
Ave. rank		11.6	15.1	16.3	9.4	7.7	13.5	15.1	6.2	13.7

Func.		CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
f_1	Mean	1.741e+07 +	6.673e+09 +	5.867e+09 +	3.910e+03 +	1.493e+03 ≈	5.211e+04 +	2.788e+08 +	**1.243e+03**
	Std	4.898e+06	1.080e+09	1.978e+09	4.392e+03	1.105e+03	7.171e+04	7.721e+08	1.231e+03
$*f_2$	Mean	2.113e+09 +	6.681e+11 +	6.501e+11 +	4.018e+05 ≈	4.249e+05 +	8.600e+06 +	6.442e+10 +	**3.222e+05**
	Std	5.010e+08	1.193e+11	2.135e+11	3.158e+05	2.822e+05	1.285e+07	1.216e+11	3.115e+05
f_3	Mean	4.956e+08 +	2.181e+11 +	2.252e+11 +	**3.458e+04** −	5.139e+04 −	9.958e+06 +	2.009e+10 +	1.605e+05
	Std	1.332e+08	2.997e+10	7.210e+10	6.908e+04	6.885e+04	1.314e+07	8.369e+10	1.427e+05
f_4	Mean	1.942e+03 +	3.650e+03 +	1.137e+04 +	**1.926e+03** −	1.928e+03 −	1.938e+03 ≈	1.943e+03 +	1.937e+03
	Std	2.572e+00	1.023e+03	1.424e+04	2.016e+00	1.979e+00	2.090e+00	1.031e+01	4.307e+00
f_5	Mean	3.714e+06 +	1.019e+07 +	1.110e+07 +	2.433e+06 +	1.877e+06 +	5.658e+06 +	3.296e+06 +	**3.415e+05**
	Std	1.013e+06	3.003e+06	8.180e+06	7.545e+05	5.942e+05	1.339e+06	2.678e+06	1.807e+05
f_6	Mean	1.282e+04 +	6.170e+05 +	4.936e+06 +	7.681e+03 +	8.122e+03 +	1.567e+04 +	2.395e+04 +	**2.180e+03**
	Std	3.095e+03	2.545e+05	2.170e+07	2.229e+03	2.326e+03	5.065e+03	1.747e+04	4.528e+02
f_7	Mean	3.453e+06 +	3.347e+07 +	3.311e+07 +	2.702e+06 +	2.629e+06 +	4.130e+06 +	3.231e+06 +	**3.158e+05**
	Std	9.340e+05	1.254e+07	2.913e+07	7.999e+05	7.264e+05	1.405e+06	3.103e+06	3.286e+05
f_8	Mean	2.412e+03 −	2.623e+03 +	5.520e+03 +	2.396e+03 −	**2.396e+03** −	2.407e+03 −	2.549e+03 +	2.440e+03
	Std	4.815e+00	1.912e+01	2.192e+03	3.751e+00	3.619e+00	2.403e+00	4.194e+01	1.719e+01
f_9	Mean	2.955e+03 +	1.074e+04 +	1.670e+04 +	2.614e+03 −	**2.614e+03** −	2.623e+03 −	6.816e+03 +	2.732e+03
	Std	9.036e+01	6.317e+02	9.523e+03	7.320e+01	7.437e+01	1.087e+02	3.197e+03	2.394e+02

Table 3 continued

Func.		CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
f_{10}	Mean	3.455e+03 +	4.367e+03 +	4.527e+03 +	3.304e+03 +	3.316e+03 +	3.337e+03 +	3.369e+03 +	**3.279e+03**
	Std	3.757e+01	1.744e+02	5.414e+02	2.513e+01	2.943e+01	2.169e+01	1.083e+02	4.359e+01
+/≈/−		9/0/1	10/0/0	10/0/0	5/1/4	5/1/4	7/1/2	10/0/0	–
Ave. rank		5.4	9.9	11.5	2.2	2.2	4.7	6.4	**2.1**

Table 4 Ablation experiments on 30-D CEC2020 benchmark functions

Func	DE		CDE		HDE		DEKI		CDEKI	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f_1	1.517e+10	2.101e+09	4.295e+03 –	3.927e+03	5.710e+09 –	1.341e+09	9.939e+09 –	1.090e+09	**2.629e+03** –	3.214e+03
f_2	1.536e+12	2.278e+11	6.697e+05 –	6.622e+05	5.026e+11 –	1.365e+11	1.097e+12 –	1.574e+11	**5.838e+05** –	6.163e+05
f_3	5.819e+11	8.297e+10	**1.489e+05** –	2.211e+05	1.715e+11 –	4.425e+10	4.084e+11 –	5.079e+10	2.806e+05 –	2.966e+05
f_4	7.402e+03	1.718e+03	1.921e+03 –	6.039e+00	2.156e+03 –	1.529e+02	4.887e+03 –	1.396e+03	**1.917e+03** –	1.786e+00
f_5	1.398e+07	4.647e+06	1.224e+05 –	5.537e+04	4.912e+06 –	2.177e+06	1.132e+07 –	4.163e+06	**1.839e+05** –	8.722e+04
f_6	1.477e+05	7.582e+04	4.418e+03 –	4.758e+03	4.217e+04 –	1.670e+04	4.231e+05 +	2.035e+05	**4.138e+03** –	3.721e+03
f_7	2.578e+07	9.071e+06	**8.103e+04** –	4.825e+04	6.258e+06 –	2.696e+06	2.566e+07 ≈	6.468e+06	2.126e+05 –	1.277e+05
f_8	2.505e+03	1.057e+01	2.407e+03 –	1.415e+01	2.460e+03 –	1.228e+01	2.524e+03 +	1.125e+01	**2.376e+03** –	9.237e+00
f_9	9.635e+03	2.713e+02	**2.622e+03** –	8.173e+01	7.284e+03 –	4.857e+02	9.572e+03 ≈	6.059e+02	2.643e+03 –	1.107e+02
f_{10}	3.654e+03	1.051e+02	2.938e+03 –	2.466e+01	3.152e+03 –	4.990e+01	3.588e+03 –	1.209e+02	**2.922e+03** –	1.809e+00
+/≈/–	–		0/0/10		0/0/10		2/2/6		0/0/10	
Ave. rank	4.8		1.6		3.0		4.2		**1.4**	

Table 5 Experimental results and statistical analyses on unconstrained HPA design tasks

Tasks			GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
HPA101	N=3, D=0	Mean	2.492e+01 +	2.523e+01 +	2.425e+01 +	2.406e+01 +	2.538e+01 +	2.577e+01 +	2.484e+01 +	2.394e+01 +	2.505e+01 +
		Std	1.908e−01	3.908e−01	3.368e−01	3.678e−01	7.774e−01	4.275e−01	8.628e−01	2.543e−01	5.360e−01
	N=3, D=1	Mean	2.452e+01 +	2.475e+01 +	2.397e+01 +	2.380e+01 +	2.492e+01 +	2.588e+01 +	2.510e+01 +	2.275e+01 +	2.454e+01 +
		Std	3.610e−01	4.696e−01	3.829e−01	4.888e−01	1.004e+00	7.570e−01	1.016e+00	2.285e−01	5.943e−01
	N=4, D=0	Mean	2.355e+01 +	2.390e+01 +	2.341e+01 +	2.304e+01 +	2.390e+01 +	2.433e+01 +	2.382e+01 +	2.267e+01 +	2.350e+01 +
		Std	1.933e−01	4.495e−01	2.737e−01	2.329e−01	6.371e−01	5.848e−01	1.113e+00	1.777e−01	3.903e−01
	N=4, D=1	Mean	2.398e+01 +	2.431e+01 +	2.418e+01 +	2.371e+01 +	2.430e+01 +	2.514e+01 +	2.526e+01 +	2.254e+01 +	2.381e+01 +
		Std	3.409e−01	5.537e−01	3.368e−01	4.075e−01	7.097e−01	7.421e−01	5.611e−01	2.232e−01	4.850e−01
	N=5, D=0	Mean	2.411e+01 +	2.439e+01 +	2.390e+01 +	2.332e+01 +	2.401e+01 +	2.498e+01 +	2.366e+01 +	2.287e+01 +	2.387e+01 +
		Std	2.096e−01	3.505e−01	3.891e−01	4.226e−01	6.103e−01	4.747e−01	9.728e−01	3.005e−01	3.048e−01
	N=5, D=1	Mean	2.466e+01 +	2.534e+01 +	2.471e+01 +	2.397e+01 +	2.500e+01 +	2.591e+01 +	2.554e+01 +	2.297e+01 +	2.452e+01 +
		Std	2.891e−01	5.725e−01	4.474e−01	5.554e−01	6.975e−01	5.718e−01	8.373e−01	2.432e−01	4.989e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			11.0	14.1	9.6	7.2	13.5	16.8	12.8	2.8	10.0
HPA102	N=3, D=0	Mean	2.292e+02 +	2.305e+02 +	2.291e+02 +	2.239e+02 +	2.289e+02 +	2.436e+02 +	2.280e+02 +	2.241e+02 +	2.281e+02 +
		Std	1.690e+00	2.798e+00	1.954e+00	1.651e+00	7.168e+00	6.892e+00	3.016e+00	1.265e+00	3.139e+00
	N=3, D=1	Mean	2.315e+02 +	2.334e+02 +	2.300e+02 +	2.255e+02 +	2.394e+02 +	2.482e+02 +	2.399e+02 +	2.217e+02 +	2.308e+02 +
		Std	2.650e+00	4.770e+00	1.933e+00	3.506e+00	1.016e+01	9.548e+00	1.005e+01	3.519e+00	5.156e+00
	N=4, D=0	Mean	2.250e+02 +	2.273e+02 +	2.243e+02 +	2.192e+02 +	2.302e+02 +	2.372e+02 +	2.258e+02 +	2.201e+02 +	2.227e+02 +
		Std	1.558e+00	2.802e+00	2.711e+00	1.618e+00	7.966e+00	8.854e+00	6.308e+00	1.972e+00	2.368e+00
	N=4, D=1	Mean	2.336e+02 +	2.388e+02 +	2.343e+02 +	2.282e+02 +	2.467e+02 +	2.449e+02 +	2.457e+02 +	2.228e+02 +	2.311e+02 +
		Std	2.859e+00	5.325e+00	3.473e+00	5.604e+00	1.227e+01	8.914e+00	1.072e+01	4.002e+00	5.303e+00
	N=5, D=0	Mean	2.241e+02 +	2.275e+02 +	2.252e+02 +	2.194e+02 +	2.234e+02 +	2.405e+02 +	2.256e+02 +	2.192e+02 +	2.244e+02 +
		Std	1.816e+00	3.531e+00	3.326e+00	2.497e+00	3.161e+00	8.290e+00	6.276e+00	2.522e+00	3.653e+00
	N=5, D=1	Mean	2.347e+02 +	2.383e+02 +	2.388e+02 +	2.306e+02 +	2.473e+02 +	2.472e+02 +	2.557e+02 +	2.259e+02 +	2.322e+02 +
		Std	2.591e+00	3.874e+00	5.345e+00	5.955e+00	1.140e+01	9.235e+00	1.019e+01	4.648e+00	4.201e+00

Table 5 continued

Tasks			GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			10.8	13.1	11.1	6.6	14.0	16.3	13.8	6.0	9.6
HPA103	N=3, D=0	Mean	− 1.110e+01 +	− 1.081e+01 +	− 1.122e+01 +	− 1.141e+01 +	− 1.130e+01 +	− 1.048e+01 +	− 1.147e+01 +	− 1.145e+01 +	− 1.122e+01 +
		Std	1.277e−01	3.302e−01	1.697e−01	1.970e−01	2.529e−01	5.657e−01	2.172e−01	2.508e−01	3.092e−01
	N=3, D=1	Mean	− 1.102e+01 +	− 1.081e+01 +	− 1.110e+01 +	− 1.138e+01 +	− 1.119e+01 +	− 1.061e+01 +	− 1.110e+01 +	− 1.143e+01 +	− 1.110e+01 +
		Std	2.033e−01	2.380e−01	2.344e−01	1.333e−01	1.611e−01	3.120e−01	2.259e−01	2.418e−01	2.381e−01
	N=4, D=0	Mean	− 1.131e+01 +	− 1.096e+01 +	− 1.126e+01 +	− 1.165e+01 +	− 1.144e+01 +	− 1.063e+01 +	− 1.159e+01 +	− 1.154e+01 +	− 1.122e+01 +
		Std	1.708e−01	3.021e−01	2.443e−01	2.227e−01	2.728e−01	5.305e−01	2.521e−01	1.763e−01	3.249e−01
	N=4, D=1	Mean	− 1.094e+01 +	− 1.068e+01 +	− 1.101e+01 +	− .112e+01 +	− 1.113e+01 +	− 1.048e+01 +	− 1.096e+01 +	− 1.142e+01 +	− 1.080e+01 +
		Std	1.417e−01	4.589e−01	2.400e−01	3.316e−01	2.663e−01	3.658e−01	2.844e−01	2.291e−01	2.991e−01
	N=5, D=0	Mean	− 1.113e+01 +	− 1.080e+01 +	− 1.112e+01 +	− 1.142e+01 +	− 1.152e+01 +	− 1.099e+01 +	− 1.167e+01 +	− 1.154e+01 +	− 1.108e+01 +
		Std	2.081e−01	3.580e−01	2.131e−01	2.387e−01	2.145e−01	3.730e−01	2.635e−01	2.165e−01	3.335e−01
	N=5, D=1	Mean	− 1.081e+01 +	− 1.050e+01 +	− 1.081e+01 +	− 1.111e+01 +	− 1.104e+01 +	− 1.057e+01 +	− 1.118e+01 +	− 1.142e+01 +	− 1.075e+01 +
		Std	1.898e−01	3.150e−01	2.445e−01	3.254e−01	3.059e−01	3.168e−01	1.808e−01	2.250e−01	3.492e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			12.3	15.3	11.6	7.6	8.3	16.0	8.2	5.8	12.6

Table 5 continued

Tasks			CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
HPA101	N=3, D=0	Mean	2.489e+01 +	2.505e+01 +	2.574e+01 +	2.392e+01 +	2.405e+01 +	2.392e+01 +	2.366e+01 +	**2.353e+01**
		Std	3.152e−01	3.732e−01	7.704e−01	1.898e−01	2.615e−01	1.439e−01	1.714e−01	8.885e−02
	N=3, D=1	Mean	2.404e+01 +	2.459e+01 +	2.569e+01 +	2.307e+01 +	2.324e+01 +	2.312e+01 +	2.283e+01 +	**2.247e+01**
		Std	3.790e−01	4.216e−01	7.799e−01	2.114e−01	2.897e−01	2.695e−01	3.387e−01	2.642e−01
	N=4, D=0	Mean	2.348e+01 +	2.368e+01 +	2.420e+01 +	2.278e+01 +	2.291e+01 +	2.281e+01 +	2.260e+01 +	**2.246e+01**
		Std	2.031e−01	2.816e−01	4.830e−01	1.235e−01	1.937e−01	1.561e−01	1.023e−01	4.418e−02
	N=4, D=1	Mean	2.345e+01 +	2.415e+01 +	2.469e+01 +	2.283e+01 +	2.292e+01 +	2.275e+01 +	2.272e+01 +	**2.211e+01**
		Std	3.542e−01	4.017e−01	8.260e−01	1.585e−01	2.083e−01	1.301e−01	3.757e−01	1.658e−01
	N=5, D=0	Mean	2.390e+01 +	2.415e+01 +	2.476e+01 +	2.308e+01 +	2.319e+01 +	2.308e+01 +	2.274e+01 +	**2.254e+01**
		Std	2.937e−01	3.132e−01	4.141e−01	1.706e−01	1.775e−01	2.391e−01	1.399e−01	7.358e−02
	N=5, D=1	Mean	2.409e+01 +	2.455e+01 +	2.578e+01 +	2.322e+01 +	2.340e+01 +	2.321e+01 +	2.306e+01 +	**2.249e+01**
		Std	3.362e−01	4.893e−01	8.543e−01	2.659e−01	3.380e−01	2.426e−01	4.835e−01	2.522e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			9.0	12.0	15.8	4.2	6.0	4.5	2.5	**1.0**
HPA102	N=3, D=0	Mean	2.278e+02 +	2.307e+02 +	2.339e+02 +	2.239e+02 +	2.238e+02 +	2.235e+02 +	2.229e+02 +	**2.218e+02**
		Std	1.604e+00	2.833e+00	5.228e+00	1.090e+00	7.569e−01	1.047e+00	1.128e+00	4.807e−01
	N=3, D=1	Mean	2.268e+02 +	2.339e+02 +	2.390e+02 +	2.217e+02 +	2.227e+02 +	2.219e+02 +	2.199e+02 +	**2.157e+02**
		Std	2.263e+00	5.036e+00	7.994e+00	2.308e+00	3.752e+00	3.784e+00	4.619e+00	3.306e+00
	N=4, D=0	Mean	2.220e+02 +	2.260e+02 +	2.281e+02 +	2.186e+02 +	2.189e+02 +	2.183e+02 +	2.176e+02 +	**2.162e+02**
		Std	1.708e+00	2.611e+00	4.997e+00	7.593e−01	1.254e+00	1.187e+00	9.715e−01	5.183e−01
	N=4, D=1	Mean	2.306e+02 +	2.362e+02 +	2.449e+02 +	2.222e+02 +	2.223e+02 +	2.214e+02 +	2.213e+02 +	**2.167e+02**
		Std	4.567e+00	4.618e+00	9.074e+00	1.973e+00	1.727e+00	1.743e+00	4.062e+00	2.065e+00
	N=5, D=0	Mean	2.214e+02 +	2.261e+02 +	2.294e+02 +	2.181e+02 +	2.181e+02 +	2.174e+02 +	2.171e+02 +	**2.153e+02**
		Std	2.075e+00	3.807e+00	8.712e+00	1.076e+00	1.049e+00	1.039e+00	1.351e+00	6.689e−01
	N=5, D=1	Mean	2.311e+02 +	2.384e+02 +	2.445e+02 +	2.244e+02 +	2.238e+02 +	2.231e+02 +	2.234e+02 +	**2.163e+02**
		Std	4.411e+00	5.447e+00	1.048e+01	3.412e+00	1.733e+00	3.013e+00	2.869e+00	2.568e+00

Table 5 continued

Tasks			CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			8.0	13.2	14.8	4.2	4.8	3.2	2.2	**1.0**
HPA103	N=3, D=0	Mean	− 1.117e+01 +	− 1.109e+01 +	− 1.054e+01 +	− 1.150e+01 +	− 1.153e+01 +	− 1.150e+01 +	− 1.164e+01 +	**− 1.184e+01**
		Std	1.549e−01	2.371e−01	4.958e−01	1.414e−01	1.102e−01	1.485e−01	1.521e−01	1.069e−01
	N=3, D=1	Mean	− 1.114e+01 +	− 1.105e+01 +	− 1.026e+01 +	− 1.154e+01 +	− 1.142e+01 +	− 1.148e+01 +	− 1.158e+01 +	**− 1.189e+01**
		Std	1.644e−01	2.870e−01	3.786e−01	1.366e−01	1.681e−01	1.644e−01	3.000e−01	1.831e−01
	N=4, D=0	Mean	− 1.140e+01 +	− 1.131e+01 +	− 1.080e+01 +	− 1.173e+01 +	− 1.170e+01 +	− 1.176e+01 +	− 1.178e+01 +	**− 1.202e+01**
		Std	1.963e−01	2.467e−01	4.774e−01	1.550e−01	1.402e−01	1.006e−01	1.830e−01	8.716e−02
	N=4, D=1	Mean	− 1.111e+01 +	− 1.100e+01 +	− 1.021e+01 +	− 1.139e+01 +	− 1.142e+01 +	− 1.139e+01 +	− 1.148e+01 +	**− 1.187e+01**
		Std	2.021e−01	2.457e−01	5.876e−01	1.807e−01	1.190e−01	2.044e−01	2.446e−01	1.568e−01
	N=5, D=0	Mean	− 1.128e+01 +	− 1.116e+01 +	− 1.062e+01 +	− 1.168e+01 +	− 1.162e+01 +	− 1.175e+01 +	− 1.180e+01 +	**− 1.203e+01**
		Std	1.695e−01	2.014e−01	5.115e−01	1.381e−01	1.513e−01	1.370e−01	1.475e−01	7.973e−02
	N=5, D=1	Mean	− 1.101e+01 +	− 1.076e+01 +	− 9.756e+00 +	− 1.144e+01 +	− 1.131e+01 +	− 1.137e+01 +	− 1.146e+01 +	**− 1.178e+01**
		Std	1.632e−01	2.386e−01	5.373e−01	1.493e−01	1.626e−01	1.524e−01	2.506e−01	1.772e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			10.0	12.3	16.6	3.8	4.8	4.3	2.0	**1.0**

Table 6 Experimental results and statistical analyses on constrained HPA design tasks

Tasks			GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
HPA131	N=3, D=0	Mean	2.531e+01 +	2.545e+01 +	2.469e+01 +	2.440e+01 +	2.558e+01 +	2.665e+01 +	2.490e+01 +	2.407e+01 +	2.531e+01 +
		Std	4.194e−01	4.878e−01	5.199e−01	4.806e−01	8.875e−01	1.027e+00	1.003e+00	3.965e−01	5.619e−01
	N=3, D=1	Mean	2.497e+01 +	2.511e+01 +	2.448e+01 +	2.437e+01 +	2.547e+01 +	2.664e+01 +	2.544e+01 +	2.294e+01 +	2.464e+01 +
		Std	4.708e−01	5.870e−01	5.945e−01	7.213e−01	1.114e+00	1.023e+00	8.195e−01	3.428e−01	6.052e−01
	N=4, D=0	Mean	2.380e+01 +	2.382e+01 +	2.381e+01 +	2.324e+01 +	2.433e+01 +	2.511e+01 +	2.399e+01 +	2.274e+01 +	2.360e+01 +
		Std	2.117e−01	3.341e−01	2.971e−01	2.804e−01	9.145e−01	8.318e−01	8.449e−01	1.459e−01	4.608e−01
	N=4, D=1	Mean	2.442e+01 +	2.463e+01 +	2.488e+01 +	2.376e+01 +	2.476e+01 +	2.603e+01 +	2.536e+01 +	2.283e+01 +	2.426e+01 +
		Std	3.404e−01	4.867e−01	5.637e−01	4.278e−01	6.734e−01	1.326e+00	5.516e−01	2.720e−01	4.740e−01
	N=5, D=0	Mean	2.446e+01 +	2.478e+01 +	2.440e+01 +	2.374e+01 +	2.437e+01 +	2.545e+01 +	2.420e+01 +	2.311e+01 +	2.425e+01 +
		Std	2.486e−01	5.016e−01	5.128e−01	3.990e−01	7.548e−01	5.596e−01	9.521e−01	2.940e−01	4.450e−01
	N=5, D=1	Mean	2.516e+01 +	2.550e+01 +	2.513e+01 +	2.446e+01 +	2.557e+01 +	2.659e+01 +	2.567e+01 +	2.312e+01 +	2.477e+01 +
		Std	4.989e−01	5.793e−01	4.543e−01	5.624e−01	8.565e−01	9.728e−01	8.246e−01	4.611e−01	5.428e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			11.0	12.8	10.3	6.7	13.8	17.0	12.3	2.4	9.1
HPA142	N=3, D=0	Mean	2.369e+02 +	2.400e+02 +	2.419e+02 +	2.305e+02 +	2.400e+02 +	2.640e+02 +	2.429e+02 +	2.313e+02 +	2.367e+02 +
		Std	3.995e+00	5.848e+00	5.240e+00	2.932e+00	1.090e+01	1.771e+01	1.610e+01	5.571e+00	3.995e+00
	N=3, D=1	Mean	2.407e+02 +	2.437e+02 +	2.489e+02 +	2.348e+02 +	2.534e+02 +	2.661e+02 +	2.611e+02 +	2.353e+02 +	2.389e+02 +
		Std	5.720e+00	6.503e+00	5.627e+00	4.784e+00	1.190e+01	1.507e+01	9.807e+00	8.582e+00	5.151e+00
	N=4, D=0	Mean	2.322e+02 +	2.327e+02 +	2.369e+02 +	2.265e+02 +	2.438e+02 +	2.633e+02 +	2.382e+02 +	2.279e+02 +	2.293e+02 +
		Std	4.683e+00	3.674e+00	4.630e+00	3.203e+00	1.826e+01	1.709e+01	1.195e+01	4.754e+00	4.777e+00
	N=4, D=1	Mean	2.434e+02 +	2.468e+02 +	2.517e+02 +	2.376e+02 +	2.595e+02 +	2.637e+02 +	2.705e+02 +	2.338e+02 +	2.402e+02 +
		Std	5.120e+00	5.843e+00	6.129e+00	5.327e+00	1.769e+01	1.229e+01	1.803e+01	5.557e+00	8.101e+00
	N=5, D=0	Mean	2.315e+02 +	2.336e+02 +	2.378e+02 +	2.266e+02 +	2.372e+02 +	2.611e+02 +	2.454e+02 +	2.284e+02 +	2.290e+02 +
		Std	3.686e+00	6.712e+00	4.170e+00	3.446e+00	1.295e+01	2.118e+01	2.101e+01	4.427e+00	3.201e+00
	N=5, D=1	Mean	2.473e+02 +	2.487e+02 +	2.553e+02 +	2.377e+02 +	2.634e+02 +	2.635e+02 +	2.756e+02 +	2.367e+02 +	2.396e+02 +
		Std	5.616e+00	6.256e+00	5.923e+00	8.312e+00	1.838e+01	1.273e+01	1.678e+01	6.455e+00	4.861e+00

Table 6 continued

Tasks			GA	PSO	DE	CMA-ES	GWO	WOA	GJO	RIME	CPSO
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			10.2	11.5	13.6	6.3	14.5	16.6	16.2	6.3	8.7
HPA143	N=3, D=0	Mean	− 1.082e+01 +	− 1.088e+01 +	− 1.095e+01 +	− 1.128e+01 +	− 1.118e+01 +	− 1.032e+01 +	− 1.133e+01 +	− 1.130e+01 +	− 1.104e+01 +
		Std	1.829e−01	2.871e−01	2.664e−01	2.056e−01	2.332e−01	6.171e−01	1.614e−01	2.610e−01	3.464e−01
	N=3, D=1	Mean	− 1.077e+01 +	− 1.068e+01 +	− 1.063e+01 +	− 1.114e+01 +	− 1.101e+01 +	− 1.040e+01 +	− 1.105e+01 +	− 1.122e+01 +	− 1.088e+01 +
		Std	2.536e−01	3.373e−01	3.457e−01	2.721e−01	2.175e−01	6.864e−01	1.788e−01	2.691e−01	2.879e−01
	N=4, D=0	Mean	− 1.105e+01 +	− 1.091e+01 +	− 1.093e+01 +	− 1.145e+01 +	− 1.138e+01 +	− 1.063e+01 +	− 1.172e+01 +	− 1.164e+01 +	− 1.131e+01 +
		Std	1.877e−01	2.595e−01	3.781e−01	2.485e−01	3.600e−01	4.419e−01	1.977e−01	1.798e−01	2.948e−01
	N=4, D=1	Mean	− 1.067e+01 +	− 1.045e+01 +	− 1.067e+01 +	− 1.096e+01 +	− 1.101e+01 +	− 9.803e+00 +	− 1.101e+01 +	− 1.122e+01 +	− 1.084e+01 +
		Std	1.999e−01	2.892e−01	2.700e−01	3.883e−01	2.788e−01	7.867e−01	1.532e−01	2.044e−01	2.660e−01
	N=5, D=0	Mean	− 1.089e+01 +	− 1.079e+01 +	− 1.083e+01 +	− 1.106e+01 +	− 1.123e+01 +	− 1.083e+01 +	− 1.168e+01 +	− 1.143e+01 +	− 1.086e+01 +
		Std	2.817e−01	3.743e−01	3.332e−01	4.430e−01	3.009e−01	6.123e−01	1.946e−01	2.736e−01	2.912e−01
	N=5, D=1	Mean	− 1.057e+01 +	− 1.030e+01 +	− 1.053e+01 +	− 1.077e+01 +	− 1.101e+01 +	− 1.022e+01 +	− 1.109e+01 +	− 1.114e+01 +	− 1.069e+01 +
		Std	1.888e−01	3.288e−01	3.088e−01	3.452e−01	3.220e−01	9.311e−01	2.266e−01	2.926e−01	4.836e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
Ave. rank			13.0	14.8	14.2	8.8	8.5	15.8	5.2	5.2	11.5

Table 6 continued

Tasks			CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
HPA131	N=3, D=0	Mean	2.551e+01 +	2.552e+01 +	2.652e+01 +	2.448e+01 +	2.460e+01 +	2.431e+01 +	2.390e+01 +	**2.362e+01**
		Std	3.328e−01	4.822e−01	9.169e−01	3.318e−01	3.464e−01	2.758e−01	4.680e−01	1.815e−01
	N=3, D=1	Mean	2.477e+01 +	2.491e+01 +	2.621e+01 +	2.363e+01 +	2.384e+01 +	2.348e+01 +	2.334e+01 +	**2.257e+01**
		Std	4.045e−01	5.014e−01	1.042e+00	3.528e−01	5.808e−01	3.545e−01	6.392e−01	2.872e−01
	N=4, D=0	Mean	2.382e+01 +	2.416e+01 +	2.462e+01 +	2.303e+01 +	2.316e+01 +	2.300e+01 +	2.289e+01 +	**2.249e+01**
		Std	2.727e−01	4.751e−01	6.675e−01	2.012e−01	1.679e−01	2.169e−01	2.037e−01	7.086e−02
	N=4, D=1	Mean	2.404e+01 +	2.447e+01 +	2.551e+01 +	2.315e+01 +	2.344e+01 +	2.311e+01 +	2.334e+01 +	**2.236e+01**
		Std	6.248e−01	5.549e−01	1.067e+00	2.198e−01	2.519e−01	2.564e−01	5.280e−01	1.739e−01
	N=5, D=0	Mean	2.424e+01 +	2.446e+01 +	2.504e+01 +	2.337e+01 +	2.342e+01 +	2.341e+01 +	2.302e+01 +	**2.267e+01**
		Std	3.410e−01	4.458e−01	7.274e−01	2.669e−01	3.116e−01	2.681e−01	2.340e−01	1.714e−01
	N=5, D=1	Mean	2.457e+01 +	2.547e+01 +	2.614e+01 +	2.376e+01 +	2.377e+01 +	2.375e+01 +	2.381e+01 +	**2.273e+01**
		Std	4.406e−01	5.655e−01	7.513e−01	3.785e−01	3.645e−01	3.069e−01	6.414e−01	3.554e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			9.8	12.6	16.0	4.6	6.0	3.8	3.5	**1.0**
HPA142	N=3, D=0	Mean	2.348e+02 +	2.391e+02 +	2.424e+02 +	2.288e+02 +	2.293e+02 +	2.287e+02 +	2.287e+02 +	**2.241e+02**
		Std	3.393e+00	5.452e+00	1.090e+01	2.066e+00	2.534e+00	2.007e+00	3.340e+00	1.209e+00
	N=3, D=1	Mean	2.393e+02 +	2.440e+02 +	2.485e+02 +	2.303e+02 +	2.317e+02 +	2.298e+02 +	2.320e+02 +	**2.207e+02**
		Std	6.155e+00	8.689e+00	1.215e+01	3.800e+00	4.481e+00	4.738e+00	7.120e+00	2.136e+00
	N=4, D=0	Mean	2.286e+02 +	2.332e+02 +	2.376e+02 +	2.242e+02 +	2.238e+02 +	2.240e+02 +	2.241e+02 +	**2.185e+02**
		Std	3.784e+00	6.170e+00	1.153e+01	2.474e+00	1.775e+00	2.454e+00	2.854e+00	1.266e+00
	N=4, D=1	Mean	2.394e+02 +	2.428e+02 +	2.558e+02 +	2.319e+02 +	2.307e+02 +	2.306e+02 +	2.347e+02 +	**2.233e+02**
		Std	5.695e+00	6.116e+00	8.628e+00	4.519e+00	4.380e+00	4.124e+00	6.192e+00	4.275e+00
	N=5, D=0	Mean	2.295e+02 +	2.358e+02 +	2.389e+02 +	2.236e+02 +	2.243e+02 +	2.229e+02 +	2.234e+02 +	**2.176e+02**
		Std	3.062e+00	6.761e+00	1.305e+01	2.184e+00	2.580e+00	1.543e+00	3.178e+00	1.450e+00
	N=5, D=1	Mean	2.391e+02 +	2.477e+02 +	2.542e+02 +	2.335e+02 +	2.359e+02 +	2.314e+02 +	2.369e+02 +	**2.211e+02**
		Std	5.174e+00	6.891e+00	1.353e+01	3.122e+00	5.156e+00	4.461e+00	5.781e+00	3.580e+00

Table 6 continued

Tasks			CLPSO	CLPSO-LS	PPSO	SHADE	iL-SHADE	Hip-DE	GTDE	CDEKI
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			8.3	11.3	14.0	3.8	3.8	2.4	4.3	**1.0**
HPA143	N=3, D=0	Mean	− 1.107e+01 +	− 1.097e+01 +	− 9.971e+00 +	− 1.143e+01 +	− 1.131e+01 +	− 1.143e+01 +	− 1.137e+01 +	**-1.174e+01**
		Std	1.680e−01	2.636e−01	5.692e−01	1.971e−01	1.870e−01	1.882e−01	2.534e−01	1.028e−01
	N=3, D=1	Mean	− 1.103e+01 +	− 1.088e+01 +	− 9.968e+00 +	− 1.133e+01 +	− 1.129e+01 +	− 1.136e+01 +	− 1.135e+01 +	**-1.177e+01**
		Std	1.493e−01	1.814e−01	5.681e−01	1.517e−01	1.876e−01	2.168e−01	3.128e−01	1.171e−01
	N=4, D=0	Mean	− 1.132e+01 +	− 1.098e+01 +	− 1.036e+01 +	− 1.160e+01 +	− 1.155e+01 +	− 1.158e+01 +	− 1.154e+01 +	**-1.193e+01**
		Std	1.189e−01	1.737e−01	5.757e−01	1.402e−01	1.844e−01	1.824e−01	1.859e−01	1.190e−01
	N=4, D=1	Mean	− 1.090e+01 +	− 1.073e+01 +	− 9.853e+00 +	− 1.134e+01 +	− 1.121e+01 +	− 1.134e+01 +	− 1.118e+01 +	**-1.172e+01**
		Std	2.011e−01	2.090e−01	5.475e−01	1.595e−01	1.829e−01	1.377e−01	2.645e−01	1.359e−01
	N=5, D=0	Mean	− 1.115e+01 +	− 1.093e+01 +	− 1.012e+01 +	− 1.148e+01 +	− 1.147e+01 +	− 1.156e+01 +	− 1.153e+01 +	**-1.188e+01**
		Std	2.082e−01	3.197e−01	6.651e−01	1.531e−01	1.776e−01	1.821e−01	2.750e−01	1.589e−01
	N=5, D=1	Mean	− 1.087e+01 +	− 1.077e+01 +	− 9.498e+00 +	− 1.119e+01 +	− 1.115e+01 +	− 1.114e+01 +	− 1.106e+01 +	**-1.162e+01**
		Std	1.410e−01	3.047e−01	5.402e−01	2.030e−01	2.041e−01	1.606e−01	3.436e−01	2.322e−01
+/≈/−			6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	–
Ave. rank			9.5	11.7	16.8	3.2	5.2	3.5	5.2	**1.0**

while, the convergence curves of representative functions on CEC2020 are shown in Fig. 3.

Ablation experiments on CEC2020 To quantitatively investigate the proposed DE/winner-to-best/1 mutation operation, hybrid local search operation, and repair operation with knowledge inheritance independently, the simple DE is employed as the basic optimizer, and each proposed strategy is integrated with simple DE to construct a new variant. The variants are listed as follows:

- CDE: simple DE + DE/winner-to-best/1 mutation operation.
- HDE: simple DE + hybrid local search operation.
- DEKI: simple DE + repair operation with knowledge inheritance.
- CDEKI: our complete proposal.

The simple DE adopts the DE/rand/1/bin strategy; while, the scaling factor F and crossover rate Cr are fixed at 0.8 and 0.8, respectively. Table 4 summarizes the results of ablation experiments on 30-D CEC2020 benchmark functions.

Performance on HPA design tasks Considering that HPA131, HPA142, and HPA143 in HPA design tasks contain constraints. In this situation, the conventional EA approaches including the proposed CDEKI cannot solve the constrained optimization problems directly. Therefore, to investigate the performance of CDEKI fairly, we equip all optimizers with the static penalty function, as expressed in Eq. (12).

$$F(R_i) = f(R_i) + w \cdot \sum_{i=1}^{m} (\max(0, g_i(R_i))) \tag{12}$$

where $F(\cdot)$ is the fitness function, $f(\cdot)$ is the objective function, and $g_i(\cdot)$ is the constraint function. w is a penalty coefficient set to $10e7$ recommended in [35].

Tables 5 and 6 summarize the experimental results and statistical analyses on unconstrained and constrained HPA design tasks, respectively. Figure 4 presents the corresponding convergence curves on representative tasks.

5 Discussion

This section begins by providing the theoretical computational complexity analysis of our proposed CDEKI. Subsequently, we analyze the performance of CDEKI on CEC2020 benchmark functions including the ablation experiments. Finally, we analyze the practical performance of CDEKI on diverse HPA design tasks.

5.1 Computational complexity analysis

Through the pseudocode of CDEKI presented in Algorithm 2, we analyze the computational complexity of CDEKI within one loop (i.e., from Algorithm 2 Line 5 to 13) initially. Supposing the population size is N, the dimension size is D, and the

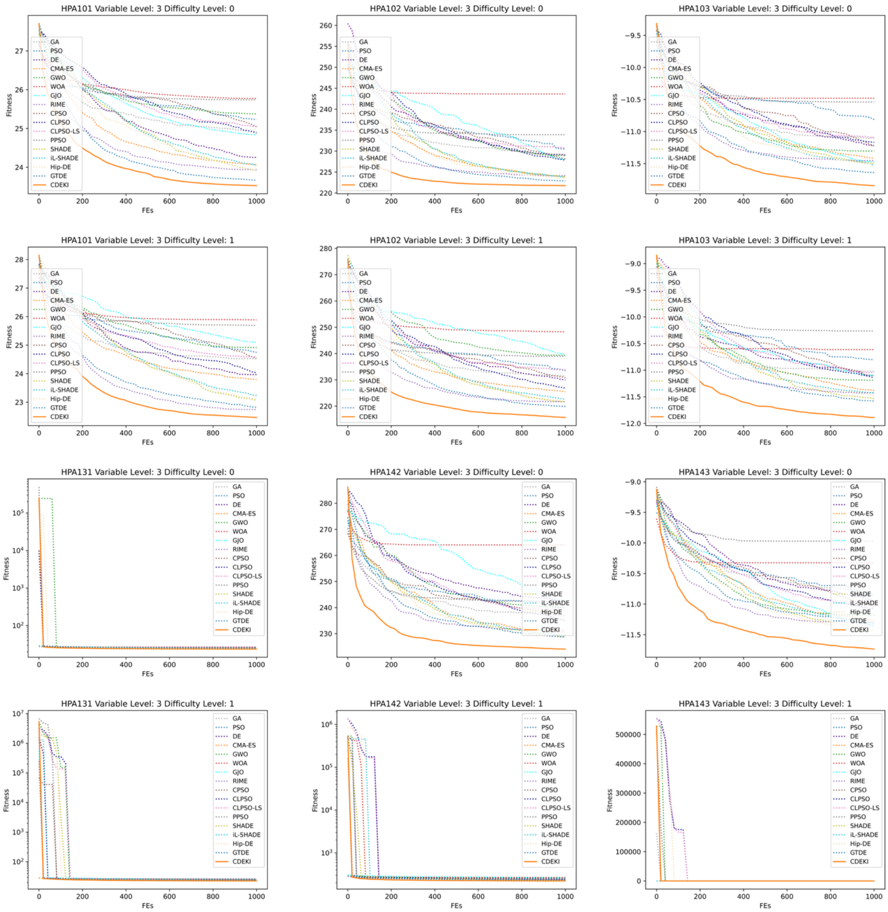


Fig. 4 Convergence curves of competitor algorithms on representative HPA design tasks

maximum iteration is T, the computational complexity of each component is summarized as follows:

- Select a random competitor $\boldsymbol{x}_{r1}^{g}$: $O(1)$.
- Construct the mutated vector $\boldsymbol{v}_{i}^{g}$ using Eq. (7): $O(D)$.
- Construct the offspring individual $\boldsymbol{u}_{i}^{g}$ using Eq. (8): $O(2D)$.
- Repair the offspring individual $\boldsymbol{u}_{i}^{g}$ using Algorithm (2): $O(D)$.
- Greedy selection using Eq. (5): $O(D)$.

Therefore, the computational complexity of the main loop is $O \cdot (N \cdot T \cdot (4D + 1))$. Additionally, the population initialization has a complexity of $O \cdot (N \cdot T \cdot D)$. In summary, the complete computational complexity of CDEKI is $O \cdot (N \cdot T \cdot (4D + 2)) := O(N \cdot T \cdot D)$, which is consistent with most EAs.

5.2 Performance analysis on CEC2020 benchmark functions

Since the CEC2022 benchmark suite contains test functions with various characteristics such as unimodal, multimodal, hybrid, and composite; thus, the optimization in these test functions can fully reflect the performance of optimizers and support us in investigating the features of involved algorithms thoroughly.

Initially, CEC2020 f_1 is an unimodal function, and the optimization in this function allows the performance evaluation in the aspect of the exploitative capacity. The superiority of CDEKI is apparent in CEC2020 f_1 compared with state-of-the-art optimizers, and this superiority is not affected as the dimension of the problem increases. Therefore, we conclude that CDEKI has a remarkable exploitation ability and robust performance across various problem domains.

Subsequently, f_2 to f_4 in CEC2020 are unimodal functions. These functions contain more than one local optima and evaluate the performance of optimizers in escaping from local optima and global convergence. Through the experimental results and statistical analyses summarized in Sect. 4.2, the competitiveness of our proposed CDEKI is observable. As state-of-the-art optimizers and advanced variants, RIME, SHADE, and iL-SHADE perform significantly better than CDEKI in some instances, the excellent performance of CDEKI cannot be neglected. Overall, CDEKI has the best performance, as evaluated by the average rank in CEC2020 across all tested dimensions, and the capacities to escape from local optima and global convergence are experimentally verified through the results.

Finally, the rest of the functions are hybrid and composite. These functions have complex fitness landscapes and multiple optima, which challenges the abilities of optimizers in balancing exploitation and exploration, avoiding premature convergence, and achieving global optimization. Upon review of the result summary, it becomes evident that CDEKI consistently demonstrates superior performance across many test functions within this category, thereby highlighting its efficacy in complex optimization environments.

5.3 Performance analysis on ablation experiments

Table 4 summarizes the results of ablation experiments. From these findings, it becomes apparent that introducing three proposed strategies (i.e., the DE/winner-to-best/1 mutation operation, hybrid local search operation, and repair operation with knowledge inheritance) effectively enhances the performance of DE while simultaneously expediting optimization convergence.

While it is worth noting that the proposed repair operation with knowledge inheritance may exhibit inefficiencies and occasional convergence stagnation in specific instances, the overall performance of DEKI surpasses that of standard DE as evaluated by the average rank metric. Furthermore, the introduction of the competitive mechanism and the innovative design of the DE/winner-to-best/1 mutation operation significantly contribute to the enhanced performance of DE, thus underscoring the success of our proposed CDEKI.

Through the complete ablation experiment, each of the three proposed strategies independently accelerates optimization. However, for optimal improvements, we recommend the combined utilization of these strategies, as they collectively yield superior enhancements in performance.

5.4 Performance analysis on HPA design tasks

The HPA design stands as a formidable real-world optimization challenge, demanding sophisticated solutions from EAs to address the complexities inherent in real-world scenarios. This study introduces CDEKI as a novel approach tailored to tackle HPA design tasks effectively. Remarkably, our proposed CDEKI outperforms all other methods across all HPA design instances, showcasing its unparalleled performance in this domain. This success can be attributed to several key factors: the innovative integration of a competitive mechanism, meticulously designed search operators, and strategic incorporation of the knowledge inheritance scheme. These elements collectively empower CDEKI to solve HPA design tasks with outstanding efficiency and effectiveness.

In summary, our proposed CDEKI emerges as a high-caliber optimizer, demonstrating exceptional capabilities in addressing real-world challenges. With its robust performance and versatile design, CDEKI holds immense promise for tackling a wide range of real-world optimization tasks.

6 Conclusion

This paper introduces a straightforward yet highly effective optimization technique derived from DE as CDEKI. CDEKI introduces novel features such as the DE/winner-to-best/1 mutation operation, hybrid local search operation, and repair operation with knowledge inheritance. We conduct a comprehensive performance evaluation of CDEKI on the CEC2020 benchmark functions across various dimensions. The experimental results and statistical analyses confirm the efficiency and effectiveness of our proposed CDEKI when compared to state-of-the-art EAs. Additionally, the ablation experiments on CEC2020 confirm that each of the three proposed operations contributes to improving the performance and accelerating optimization convergence. Furthermore, we extend the application of CDEKI to address a real-world challenge: HPA design. The results demonstrate the practical efficiency of CDEKI across all optimization instances in this domain.

In conclusion, our proposed CDEKI exhibits significant potential as a powerful optimizer in real-world scenarios. In future research, we plan to further develop CDEKI and leverage its capabilities to address complex tasks across various application domains.

Acknowledgements This work was supported by JSPS KAKENHI Grant Number 21A402 and 24K15098 and JST SPRING Grant Number JPMJSP2119.

Author contributions Rui Zhong contributed to conceptualization, methodology, investigation, writing—original draft, Writing—review and editing, and funding acquisition. Yang Cao contributed to methodology, formal analysis, and writing—review and editing. Enzhi Zhang contributed to methodology, formal analysis, and writing—review and editing. Masaharu Munetomo contributed to writing—review and editing and project administration.**Data availability** The source code of this research code can be downloaded from https://github.com/RuiZhong961230/CDEKI.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Storn R, Price K (1997) Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11:341–359. https://doi.org/10.1023/A:1008202821328
2. Li Y, Wang S, Yang H, Chen H, Yang B (2023) Enhancing differential evolution algorithm using leader-adjoint populations. Inf Sci 622:235–268. https://doi.org/10.1016/j.ins.2022.11.106
3. Ren C, Song Z, Meng Z (2024) Differential evolution with fitness-difference based parameter control and hypervolume diversity indicator for numerical optimization. Eng Appl Artif Intell 133:108081. https://doi.org/10.1016/j.engappai.2024.108081
4. Yang Q, Yan J-Q, Gao X-D, Xu D-D, Lu Z-Y, Zhang J (2022) Random neighbor elite guided differential evolution for global numerical optimization. Inf Sci 607:1408–1438. https://doi.org/10.1016/j.ins.2022.06.029
5. Zhong R, Yu J (2024) A novel evolutionary status guided hyper-heuristic algorithm for continuous optimization. Cluster Comput. https://doi.org/10.1007/s10586-024-04593-2
6. Wu S-H, Zhan Z-H, Tan KC, Zhang J (2023) Transferable adaptive differential evolution for many-task optimization. IEEE Trans Cybern 53(11):7295–7308. https://doi.org/10.1109/TCYB.2023.3234969
7. Yang X, Wang R, Zhao D, Yu F, Heidari AA, Xu Z, Chen H, Algarni AD, Elmannai H, Xu S (2023) Multi-level threshold segmentation framework for breast cancer images using enhanced differential evolution. Biomed Signal Process Control 80:104373. https://doi.org/10.1016/j.bspc.2022.104373
8. Zhong R, Yu J (2024) Hybrid remora crayfish optimization for engineering and wireless sensor network coverage optimization. Cluster Comput. https://doi.org/10.1007/s10586-024-04593-2
9. Xue Z, Yao S, Ma H, Zhang C, Zhang K, Chen Z (2023) Thermo-economic optimization of an enhanced geothermal system (EGS) based on machine learning and differential evolution algorithms. Fuel 340:127569. https://doi.org/10.1016/j.fuel.2023.127569
10. Zhang X, Zhong C, Abualigah L (2022) Foreign exchange forecasting and portfolio optimization strategy based on hybrid-molecular differential evolution algorithms. Soft Comput 27:1–19. https://doi.org/10.1007/s00500-022-07526-6
11. Zhong R, Zhang E, Munetomo M (2023) Cooperative coevolutionary surrogate ensemble-assisted differential evolution with efficient dual differential grouping for large-scale expensive optimization problems. Complex Intell Syst. https://doi.org/10.1007/s40747-023-01262-6
12. Agrawal S, Tiwari A, Yaduvanshi B, Rajak P (2023) Differential evolution with nearest better clustering for multimodal multiobjective optimization. Appl Soft Comput 148:110852. https://doi.org/10.1016/j.asoc.2023.110852
13. Biswas R, Sharma D (2023) A single-loop reliability-based design optimization using adaptive differential evolution. Appl Soft Comput 132:109907. https://doi.org/10.1016/j.asoc.2022.109907
14. Wahab MHH, Abdul Hamid NAW, Subramaniam S, Latip R, Othman M (2024) Decomposition-based multi-objective differential evolution for extractive multi-document automatic text summarization. Appl Soft Comput 151:110994. https://doi.org/10.1016/j.asoc.2023.110994
15. Namura N (2024) Single and multi-objective optimization benchmark problems focusing on human-powered aircraft design
16. Meng Z, Chen Y (2023) Differential evolution with exponential crossover can be also competitive on numerical optimization. Appl Soft Comput 146:110750. https://doi.org/10.1016/j.asoc.2023.110750
17. Ghosh A, Das S, Mallipeddi R, Das AK, Dash SS (2017) A modified differential evolution with distance-based selection for continuous optimization in presence of noise. IEEE Access 5:26944–26964. https://doi.org/10.1109/ACCESS.2017.2773825

18. Qin AK, Suganthan PN (2005) Self-adaptive differential evolution algorithm for numerical optimization. In: 2005 IEEE Congress on Evolutionary Computation, vol. 2, pp. 1785–17912. https://doi.org/10.1109/CEC.2005.1554904
19. Thieu NV (2024) Opfunu: an open-source python library for optimization benchmark functions. J Open Res Softw. https://doi.org/10.5334/jors.508
20. Srinivas M, Patnaik LM (1994) Genetic algorithms: a survey. Computer 27(6):17–26. https://doi.org/10.1109/2.294849
21. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, pp. 1942–19484 . https://doi.org/10.1109/ICNN.1995.488968
22. Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159–195. https://doi.org/10.1162/106365601750190398
23. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
24. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008
25. Chopra N, Mohsin Ansari M (2022) Golden jackal optimization: A novel nature-inspired optimizer for engineering applications. Expert Syst Appl 198:116924. https://doi.org/10.1016/j.eswa.2022.116924
26. Su H, Zhao D, Heidari AA, Liu L, Zhang X, Mafarja M, Chen H (2023) Rime: A physics-based optimization. Neurocomputing 532:183–214. https://doi.org/10.1016/j.neucom.2023.02.010
27. Tanabe R, Fukunaga A (2013) Success-history based parameter adaptation for differential evolution. In: 2013 IEEE Congress on Evolutionary Computation, pp. 71–78. https://doi.org/10.1109/CEC.2013.6557555
28. Brest J, Maučec MS, Bošković B (2016) il-shade: Improved l-shade algorithm for single objective real-parameter optimization. In: 2016 IEEE Congress on Evolutionary Computation (CEC), pp. 1188–1195 . https://doi.org/10.1109/CEC.2016.7743922
29. Meng Z, Yang C (2021) Hip-de: Historical population based mutation strategy in differential evolution with parameter adaptive mechanism. Inf Sci 562:44–77. https://doi.org/10.1016/j.ins.2021.01.031
30. Wang Z-J, Jian J-R, Zhan Z-H, Li Y, Kwong S, Zhang J (2023) Gene targeting differential evolution: A simple and efficient method for large-scale optimization. IEEE Trans Evol Comput 27(4):964–979. https://doi.org/10.1109/TEVC.2022.3185665
31. Chuang L-Y, Hsiao C-J, Yang C-H (2011) Chaotic particle swarm optimization for data clustering. Expert Syst Appl 38(12):14555–14563. https://doi.org/10.1016/j.eswa.2011.05.027
32. Liang JJ, Qin AK, Suganthan PN, Baskar S (2006) Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Trans Evol Comput 10(3):281–295. https://doi.org/10.1109/TEVC.2005.857610
33. Cao Y, Zhang H, Li W, Zhou M, Zhang Y, Chaovalitwongse WA (2019) Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions. IEEE Trans Evol Comput 23(4):718–731. https://doi.org/10.1109/TEVC.2018.2885075
34. Ghasemi M, Akbari E, Rahimnejad A, Razavi SE, Ghavidel S, Li L (2019) Phasor particle swarm optimization: a simple and efficient variant of PSO. Soft Comput 23(19):9701–9718. https://doi.org/10.1007/s00500-018-3536-8
35. Zhong R, Yu J, Zhang C, Munetomo M (2024) Srime: a strengthened rime with Latin hypercube sampling and embedded distance-based selection for engineering optimization problems. Neural Comput Appl. https://doi.org/10.1007/s00521-024-09424-4

Authors and Affiliations

Rui Zhong[1] · Yang Cao[2] · Enzhi Zhang[2] · Masaharu Munetomo[1]

✉ Rui Zhong
zhongrui@iic.hokudai.ac.jp

Yang Cao
yang.cao.y4@elms.hokudai.ac.jp

Enzhi Zhang
enzhi.zhang.n6@elms.hokudai.ac.jp

Masaharu Munetomo
munetomo@iic.hokudai.ac.jp

[1] Information Initiative Center, Hokkaido University, Sapporo, Japan

[2] Graduate School of Information Science and Technology, Hokkaido University, Sapporo, Japan