Hierarchical RBF Neural Network Ensemble with Adaptive Global-Local Fusion for Data-Driven Expensive Optimization

Qian-Xia Jing
College of Cryptology and Cyber Science
Nankai University
Tianjin, China

Jian-Yu Li
College of Artificial Intelligence
Nankai University
Tianjin, China
jianyulics@foxmail.com

Jun Zhang
Hanyang University
Ansan, South Korea

***Abstract*—Data-driven evolutionary algorithms (DDEAs) have emerged as effective approaches for solving expensive optimization problems (EOPs), where radial basis function neural networks (RBFNNs) are widely used as surrogate models due to their strong function approximation capability. However, existing RBFNN-based DDEAs typically rely on global surrogate models only, which may lose local information when handling medium-to-high dimensional complex multimodal functions. To address this limitation, this paper proposes HCLS-DDEA, a hierarchical clustering local surrogate DDEA with adaptive global-local fusion. HCLS-DDEA first partitions the decision space into multiple local subspaces by K-Means, then constructs a two-layer large-scale RBFNN ensemble in which the global layer captures the overall fitness landscape and the local layer fits high-frequency local features. In addition, model quality is evaluated by Bagging with OOB/CV errors, and high-quality models are dynamically selected through a roulette wheel mechanism. Finally, an adaptive weighted fusion predictor is designed based on distance and quality factors to smoothly coordinate global and local predictions. Experimental results on five benchmark functions in 10, 30, and 50 dimensions demonstrate that HCLS-DDEA significantly outperforms existing methods, particularly on 30-dimensional problems.**

***Index Terms*—RBFNN, ensemble learning, hierarchical neural network architecture, adaptive fusion prediction, expensive optimization problem, data-driven evolutionary algorithm**

I. Introduction

Data-driven evolutionary algorithms (DDEAs) are effective approaches for expensive optimization problems (EOPs) [1], [2], where objective evaluations are costly and surrogate models are introduced to reduce the number of real evaluations.

Improvements to DDEAs mainly focus on sample acquisition and surrogate modeling [1], [3]. Among surrogate models, radial basis function neural networks (RBFNNs) are widely used because of their strong nonlinear approximation ability [4], [5]. Ensemble learning has further been introduced to improve robustness by reducing the bias and variance of single models [1], [6].

However, existing RBFNN-based DDEAs usually employ only global surrogate models, which may miss local information on medium-to-high dimensional multimodal functions. To address this issue, this paper proposes HCLS-DDEA, a hierarchical clustering local surrogate DDEA with adaptive global-local fusion. The main contributions are: 1) a K-Means-based decision-space partitioning strategy for hierarchical RBFNN training; 2) a two-layer RBFNN ensemble architecture in which the global layer captures overall trends and the local layer fits high-frequency details; 3) a quality evaluation and roulette-wheel dynamic selection mechanism using OOB error for global models and purified five-fold CV error for local models; and 4) an adaptive weighted fusion predictor based on distance and quality factors.

II. Background and Related Work

A. Data-Driven Evolutionary Algorithms

Unlike traditional evolutionary algorithms, DDEAs use sampled data to build surrogate models of the expensive objective function. They are generally divided into offline DDEAs, which rely on pre-collected data [7], and online DDEAs, which update models during optimization [8].

B. Offline Data-Driven Evolutionary Optimization

Offline DDEAs commonly employ Gaussian processes (GP) [9], RBF models [4], [5], and SVMs [10] as surrogate models.

To improve robustness, ensemble learning has also been introduced. For example, DDEA-SE [6], [11] uses Bagging to construct RBF model pools, and BDDEA-LDG [6], [12] uses Boosting to improve model accuracy. However, global ensemble methods still treat the decision space uniformly and may miss high-frequency local structures in complex multimodal problems.

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62406152, in part by the Natural Science Foundation of Tianjin under Grant 24JCQNJC02100, in part by the Tianjin Top Scientist Studio Project under Grant 24JRRCRC00030, in part by the Tianjin Belt and Road Joint Laboratory under Grant 24PTLYHZ00250, in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.RS-2025-00555463, and RS-2025-25456394), and in part by the China Higher Education Institution Industry-University-Research Innovation Fund under Grant 2025XJS008. Corresponding authors: Jian-Yu Li

C. Local Surrogate Modeling and Clustering Methods

To improve local fitting, researchers have adopted divide-and-conquer strategies that partition the decision space by clustering methods such as K-Means [13] and train local models within each subspace. Methods such as SA-COSO [14] follow this idea to improve fitting accuracy.

However, existing methods still face two challenges:

1) Data Sparsity Problem: In high-dimensional scenarios, available samples are limited. When partitioned into subregions, samples in each cluster may be insufficient for quality model training, leading to overfitting.

2) Boundary Discontinuity Problem: Most methods adopt hard-switching strategies, which may introduce artificial discontinuities at cluster boundaries and mislead the search process.

D. Research Motivation for HCLS-DDEA

Based on the above analysis, a unified framework is needed to combine the generalization ability of global ensembles with the fitting accuracy of local models. HCLS-DDEA is developed for this purpose, using adaptive soft fusion to mitigate local overfitting, reduce boundary discontinuity, and balance global exploration with local exploitation.

III. HCLS-DDEA Algorithm

A. Algorithm Framework

To address the problem of insufficient local information learning, this paper develops HCLS-DDEA, a hierarchical clustering local surrogate DDEA based on a hierarchical RBFNN ensemble. The algorithm comprises the following four core modules:

1) K-Means-based Space Partitioning: The K-Means clustering algorithm is employed to partition the decision space into K local subspaces, providing data grouping for hierarchical RBFNN training, and computing the center c_k and characteristic radius R_k of each cluster.

2) Two-layer Large-scale RBFNN Ensemble Training: The first layer trains M_{global} global RBFNNs to capture the overall topological trends of the fitness function; the second layer trains M_{local} local RBFNNs on each clustered subspace C_k to finely fit high-frequency local features. Each RBFNN uses Gaussian radial basis functions as hidden-layer activation functions.

3) Model Quality-based Neural Network Model Selection: The Bagging strategy is adopted to train neural network ensembles. OOB samples are used to evaluate global RBFNNs, while purified five-fold CV errors are used to evaluate local RBFNNs. High-quality model subsets are dynamically selected from the neural network pool through a roulette wheel mechanism in each generation of evolution.

4) Adaptive Weighted Fusion Prediction: Based on the distance from the candidate solution to the nearest cluster center and model quality, the fusion weights of global and local RBFNN predictions are dynamically computed to achieve a smooth transition.

B. K-Means-based Space Partitioning

To achieve decomposition of the high-dimensional decision space, we first employ the K-Means clustering algorithm to partition the sample points in the offline dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ into K local subspaces. Traditional K-Means clustering treats all data equally, tending to position cluster centers at the geometric center of the space rather than near the function optimum. To address this, we propose a **Top-K Guided Clustering Strategy** that uses only high-quality samples to locate cluster centers:

1) Sort the dataset by fitness value and select the top $\alpha \times N$ high-quality samples ($\alpha = 0.5$)
2) Perform K-Means clustering only on high-quality samples to obtain K cluster centers $\{\mathbf{c}_k\}_{k=1}^K$
3) Assign all samples (including low-quality ones) to the nearest cluster center

The clustering objective is to minimize within-cluster variance:

$$J = \sum_{k=1}^{K} \sum_{\mathbf{x} \in C_k^{top}} \|\mathbf{x} - \mathbf{c}_k\|^2 \tag{1}$$

where C_k^{top} represents the subset of high-quality samples in the k-th cluster. This strategy makes cluster centers more likely to fall near local optima (valleys), allowing local RBFNNs to focus on promising regions. To ensure clustering stability, K-Means is run 10 times and the best result is selected.

To adapt to optimization problems of different dimensions, the number of clusters K adopts a dimension-adaptive strategy:

$$K = \max(2, \lfloor d/10 \rfloor) \tag{2}$$

After clustering is completed, the characteristic radius R_k of each cluster is computed for subsequent adaptive fusion:

$$R_k = \frac{1}{|C_k|} \sum_{\mathbf{x} \in C_k} \|\mathbf{x} - \mathbf{c}_k\| \tag{3}$$

C. Two-layer Large-scale Ensemble Model Training

1) First Layer: Global Ensemble Model Pool: The global layer aims to eliminate single-model bias through large-scale ensembling and capture the overall topological structure of the fitness function. The Bagging strategy is used to train M_{global} RBFNN models:

For $i = 1, \ldots, M_{global}$, the training process of the i-th global model G_i is as follows:

1) Randomly sample γ_{global} proportion of samples from dataset $\mathcal{D}$ to form training set $\mathcal{D}_i^{train}$
2) Use $\mathcal{D}_i^{train}$ to train RBFNN model G_i
3) Compute OOB error on out-of-bag samples $\mathcal{D}_i^{OOB} = \mathcal{D} \setminus \mathcal{D}_i^{train}$:

$$\epsilon_{OOB}^{G_i} = \frac{1}{|\mathcal{D}_i^{OOB}|} \sum_{(\mathbf{x},y) \in \mathcal{D}_i^{OOB}} (G_i(\mathbf{x}) - y)^2 \tag{4}$$

The number of global models is set to $M_{global} = 100 \times d$, where d is the problem dimension. Through large-scale ensembling, prediction variance can be effectively reduced, providing robust global predictions in data-sparse regions.

2) Second Layer: Local Ensemble Model Pool: The local layer aims to finely reconstruct high-frequency details that are smoothed out by global models. For each cluster C_k, M_{local} local RBFNN models are trained independently. To simultaneously ensure local fitting accuracy and boundary smoothness, we propose a **Hybrid Data Strategy**:

For the k-th cluster, construct hybrid training dataset $\mathcal{D}_k^{hybrid}$:

$$\mathcal{D}_k^{hybrid} = \mathcal{D}_k^{local} \cup \mathcal{D}_k^{global} \tag{5}$$

where:

- $\mathcal{D}_k^{local} = \{(\mathbf{x}, y) | \mathbf{x} \in C_k\}$: all samples within cluster k (approximately 70%)
- $\mathcal{D}_k^{global}$: randomly sample $0.3N$ samples from other clusters (approximately 30%)

This hybrid strategy has dual advantages: (1) intra-cluster data ensures local feature learning; (2) global sampling smooths cluster boundaries, preventing local neural networks from failing at boundary predictions.

For $j = 1, \ldots, M_{local}$, the j-th local RBFNN $L_{k,j}$ is trained using the Bagging strategy to sample γ_{local} proportion of data from $\mathcal{D}_k^{hybrid}$. To accurately evaluate the fitting quality of local RBFNNs, we adopt a **Five-Fold Cross-Validation Purified Scoring Mechanism**:

1) Partition hybrid dataset $\mathcal{D}_k^{hybrid}$ into 5 folds
2) For each fold, let $\mathcal{D}_{val}$ denote the held-out validation subset of $\mathcal{D}_k^{hybrid}$, and use the remaining four folds for training
3) Train the corresponding RBFNN and evaluate it on $\mathcal{D}_{val}$
4) **Key**: Since the local model is intended to measure fitting quality within cluster C_k, the validation error is computed *only* on the intra-cluster samples contained in $\mathcal{D}_{val}$, i.e., $\mathcal{D}_k^{local} \cap \mathcal{D}_{val}$:

$$\epsilon_{CV}^{L_{k,j}} = \frac{1}{|\mathcal{D}_k^{local} \cap \mathcal{D}_{val}|} \sum_{(\mathbf{x},y) \in \mathcal{D}_k^{local} \cap \mathcal{D}_{val}} (L_{k,j}(\mathbf{x}) - y)^2 \tag{6}$$

5) Compute 5-fold average error as model quality indicator

This purified scoring mechanism excludes interference from global data, truly reflecting the fitting capability of local neural networks within clusters, providing reliable basis for subsequent quality-driven fusion.

The number of local models is set to $M_{local} = 5 \times d$, where d is the problem dimension. Through dimension-adaptive large-scale ensembling, this hierarchical strategy decomposes the complex global regression problem into K simpler local subproblems.

D. Model Quality-based Dynamic Model Selection

To select high-quality models for prediction in each generation of evolution, we propose a roulette wheel selection mechanism based on model quality indicators.

First, convert the corresponding model quality error to model fitness:

$$fitness_i = \frac{1}{\epsilon^{(i)} + \varepsilon} \tag{7}$$

where $\epsilon^{(i)}$ denotes $\epsilon_{OOB}^{G_i}$ for a global model and $\epsilon_{CV}^{L_{k,j}}$ for a local model, and ε is a small constant to prevent division by zero.

Then, model subsets are selected from the model pool through roulette wheel selection. The probability of each model being selected is:

$$P(i) = \frac{fitness_i}{\sum_j fitness_j} \tag{8}$$

At the beginning of each generation, M_{global}^{select} models are selected from the global model pool, and M_{local}^{select} models are selected from each local model pool respectively. This dynamic selection mechanism allows high-quality models to have a higher probability of being selected while maintaining diversity.

E. Adaptive Weighted Fusion Prediction Mechanism

For a new candidate solution $\mathbf{x}_{new}$, we design an adaptive weighted fusion mechanism based on distance and quality factors to address the boundary discontinuity problem caused by hard-switching.

1) Global Prediction: Use the selected global model subset to compute the global prediction value:

$$\hat{y}_{global}(\mathbf{x}_{new}) = \frac{1}{M_{global}^{select}} \sum_{i \in S_{global}} G_i(\mathbf{x}_{new}) \tag{9}$$

2) Local Localization and Prediction: First compute the distance from $\mathbf{x}_{new}$ to each cluster center, and determine the nearest cluster k^*:

$$k^* = \arg\min_k \|\mathbf{x}_{new} - \mathbf{c}_k\| \tag{10}$$

Define the minimum distance:

$$d_{min} = \|\mathbf{x}_{new} - \mathbf{c}_{k^*}\| \tag{11}$$

Then use the local model subset of cluster k^* to compute the local prediction value:

$$\hat{y}_{local}(\mathbf{x}_{new}) = \frac{1}{M_{local}^{select}} \sum_{j \in S_{local}^{k^*}} L_{k^*,j}(\mathbf{x}_{new}) \tag{12}$$

3) Adaptive Fusion: The fusion weights are jointly determined by distance and quality factors:

Distance Threshold Check: If $d_{min} > \alpha \cdot R_{k^*}$, then use global prediction directly, where α is the distance threshold factor.

Distance Factor: Use Gaussian kernel to compute distance weight:

$$w_{distance} = \exp\left(-\frac{d_{min}^2}{2R_{k^*}^2}\right) \tag{13}$$

Quality Factor: Use Sigmoid function to adjust weight according to model quality ratio:

$$w_{quality} = \frac{1}{1 + \exp(-\beta \cdot (r_{quality} - 1))} \tag{14}$$

Algorithm 1 HCLS-DDEA Algorithm

Input: Offline dataset $\mathcal{D}$, dimension d, bounds $[\mathbf{lb}, \mathbf{ub}]$
Output: Optimal solution $\mathbf{x}^*$

```
// Model Training Phase
Perform K-Means clustering to obtain K clusters with centers {c_k} and radii {R_k}
Train global ensemble model pool {G_i}_{i=1}^{M_global}, compute OOB errors
Train local ensemble model pools {L_{k,j}} for each cluster, compute 5-fold CV errors
// Evolutionary Optimization Phase
Initialize population POP using LHS
for g = 1 to g_max do
    Select global model subset S_global via roulette wheel selection
    Select local model subsets {S^k_local} for each cluster via roulette wheel selection
    Evaluate population using hybrid predictor: Y = Hybrid_Predictor(POP)
    Generate offspring via SBX crossover: NPOP_1 = SBX(POP)
    Generate offspring via polynomial mutation: NPOP_2 = Mutation(POP)
    Evaluate offspring: Y_1, Y_2 = Hybrid_Predictor(NPOP_1, NPOP_2)
    Merge populations and select top n individuals by predicted fitness
end for
return Best individual x*
```

where $r_{quality} = \bar{\epsilon}_{global}/\bar{\epsilon}_{local}^{k^*}$ is the ratio of average global OOB error to average local cross-validation error, and β controls the slope of the Sigmoid function.

Combined Weight and Final Prediction:

$$w_{local} = w_{distance} \cdot w_{quality} \cdot k_{max} \tag{15}$$

$$\hat{y}_{final}(\mathbf{x}_{new}) = w_{local} \cdot \hat{y}_{local} + (1 - w_{local}) \cdot \hat{y}_{global} \tag{16}$$

where k_{max} is the local maximum weight coefficient, controlling the maximum influence of local models.

This soft fusion mechanism dynamically adjusts weights based on the spatial position of candidate solutions and model quality, relying more on local models near cluster centers and smoothly transitioning to global models in boundary regions, effectively eliminating the fitness landscape discontinuity problem caused by hard-switching.

F. Evolutionary Optimization Flow

HCLS-DDEA employs a standard genetic algorithm framework for optimization search. Each individual is encoded as a real-valued vector $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ in the d-dimensional decision space, where each decision variable is bounded by the corresponding lower and upper search limits. Algorithm 1 presents the complete pseudocode.

TABLE I
TEST FUNCTIONS

No.	Function Name	Search Range	Optimal Value
F1	Ellipsoid	$[-5.12, 5.12]^d$	0
F2	Rosenbrock	$[-2.048, 2.048]^d$	0
F3	Ackley	$[-32.768, 32.768]^d$	0
F4	Griewank	$[-600, 600]^d$	0
F5	Rastrigin	$[-5, 5]^d$	0

Population initialization uses Latin Hypercube Sampling (LHS) to ensure uniform distribution of initial solutions. Evolutionary operators use Simulated Binary Crossover (SBX, $\eta_c = 15$, $p_c = 1.0$) and polynomial mutation ($\eta_m = 15$, $p_m = 1/d$). The selection strategy uses elitism, merging parents and offspring and ranking by predicted fitness, selecting the top n best individuals for the next generation.

IV. Experimental Studies

A. Experimental Settings

1) Test Functions: To comprehensively evaluate the performance of HCLS-DDEA, this paper selects 5 classical benchmark test functions for experiments, as shown in Table I. These functions cover unimodal functions (Ellipsoid), multimodal functions (Ackley, Rastrigin, Griewank), and ill-conditioned functions (Rosenbrock), with different characteristics and difficulty levels.

2) Offline Dataset Construction: Following the standard settings for offline data-driven optimization, Latin Hypercube Sampling (LHS) is used to generate offline datasets. The dataset size is set to $N = 11 \times d$, where d is the problem dimension. This setting simulates the data-scarce scenario in practical expensive optimization problems. The offline sample size $N = 11 \times d$ was selected as a practical data-scarce setting based on preliminary pilot experiments, so that the sample budget scales linearly with the problem dimension while remaining within a limited offline evaluation budget.

3) Algorithm Parameter Settings: The parameter settings of HCLS-DDEA are shown in Table II. Most parameters adopt dimension-adaptive strategies to adapt to optimization problems of different scales. The hyperparameters in Table II were selected based on preliminary pilot experiments to balance surrogate accuracy, ensemble diversity, and computational cost, rather than through exhaustive tuning. In particular, the dimension-dependent settings (e.g., $100\times d$, $5\times d$, and d) scale the model pool and model selection sizes with the problem dimension, so that the ensemble retains sufficient modeling capacity as the search space grows. The fusion parameters α, β, and k_{max} were fixed at moderate values to keep local influence conservative and maintain smooth transitions between local and global predictions near cluster boundaries.

4) Comparison Algorithms: To verify the effectiveness of HCLS-DDEA, the following four state-of-the-art data-driven evolutionary algorithms are selected for comparison:

- **BDDEA-LDG**: Boosting-based data-driven evolutionary algorithm with localized data generation [12], which progressively improves model accuracy through Boosting

TABLE II
HCLS-DDEA PARAMETER SETTINGS

Parameter	Symbol	Setting
Dimension-adaptive Parameters		
Global model pool size	M_{global}	$100 \times d$
Local model pool size	M_{local}	$5 \times d$
Global selection count	M_{global}^{select}	$5 \times d$
Local selection count	M_{local}^{select}	d
Number of clusters	K	See Equation (2)
Bagging Sampling Parameters		
Global sampling ratio	γ_{global}	0.5
Local sampling ratio	γ_{local}	0.8
Fusion Parameters		
Local maximum weight coefficient	k_{max}	0.15
Distance threshold factor	α	0.7
Sigmoid slope	β	3.0

- **DDEA-PES**: Data-driven evolutionary algorithm based on prediction error selection [15], which dynamically selects models through prediction error
- **CC-DDEA**: Cooperative coevolution-based data-driven evolutionary algorithm [16], which adopts decomposition strategy to handle high-dimensional problems
- **ESCO**: Data-driven algorithm based on evolutionary sampling and cooperative optimization [17], which combines sampling strategy with surrogate model optimization

5) Experimental Environment and Statistical Methods: All experiments are conducted in MATLAB R2025b environment. Each test configuration is independently run 20 times, and the true function values of the optimal solutions are recorded. Wilcoxon rank-sum test ($\alpha = 0.05$) is used for statistical significance analysis.

B. Comparison with Other Algorithms

To comprehensively evaluate the performance of HCLS-DDEA, this section compares it with four state-of-the-art data-driven evolutionary algorithms: BDDEA-LDG [12], DDEA-PES [15], CC-DDEA [16], and ESCO [17]. Table III presents the detailed comparison results on 5 test functions across 3 different dimensions (10, 30, 50). Here, "+" indicates that HCLS-DDEA is significantly better than the comparison algorithm, "≈" indicates no significant difference between the two, and "-" indicates that HCLS-DDEA is significantly worse than the comparison algorithm.

From Table III, the following conclusions can be drawn:

1) HCLS-DDEA performs most outstandingly on 30-dimensional problems, achieving 5-0-0 (five wins, zero ties, and zero losses over the five test functions) against BDDEA-LDG, 4-1-0 against DDEA-PES, 4-0-1 against CC-DDEA, and 5-0-0 against ESCO. This result validates the effectiveness of HCLS-DDEA's hierarchical local modeling strategy on medium-dimensional problems.

2) On 10-dimensional problems, HCLS-DDEA shows comparable performance to BDDEA-LDG and DDEA-PES, while outperforming CC-DDEA and ESCO. In low-dimensional cases, global models already provide good fitting, with limited additional benefits from local models.

3) Overall, combining results across three dimensions, HCLS-DDEA achieves 8-6-1 against BDDEA-LDG, 7-7-1 against DDEA-PES, 12-0-3 against CC-DDEA, and 13-1-1 against ESCO, demonstrating clear advantages on medium-dimensional expensive optimization problems.

C. Ablation Study Analysis

To verify the effectiveness of local ensemble models in HCLS-DDEA, this paper designs ablation experiments comparing the complete HCLS-DDEA with a variant using only global ensemble models (denoted as Global-Only). The Global-Only variant removes the K-Means clustering-based local modeling component and retains only the global RBFNN ensemble model for fitness prediction.

Table IV presents the comparison results between HCLS-DDEA and Global-Only on 10-dimensional and 30-dimensional test functions.

From the ablation study results:

1) Local modeling effectiveness shows dimension dependency. HCLS-DDEA achieves 2-3-0 on 10D and 1-4-0 on 30D compared to Global-Only, with advantages on specific functions (F2, F4 on 10D; F3 on 30D). On the Ackley function (F3), local models demonstrate significant advantages on 30D problems, successfully reducing average fitness from 5.01 to 4.46.

2) The adaptive fusion mechanism has good self-regulation capability. When global models perform better on certain functions, the mechanism automatically reduces local model influence, avoiding additional prediction errors and ensuring robustness across different scenarios.

3) The overall advantage comes from multi-component synergy: two-layer ensemble architecture, large-scale model pool, OOB quality evaluation, and adaptive fusion mechanism work together to achieve superior performance on medium-dimensional problems.

V. CONCLUSION

This paper proposes HCLS-DDEA for medium-to-high dimensional expensive optimization problems. The method combines K-Means-based space partitioning, a two-layer RBFNN ensemble, quality-based model selection, and adaptive global-local fusion to improve local information learning while maintaining prediction smoothness.

Experiments on five test functions across 10, 30, and 50 dimensions show that HCLS-DDEA significantly outperforms the comparison algorithms, especially on 30-dimensional problems. Future work will further study parameter sensitivity, more advanced clustering methods, diverse base learners, and extensions to multi-objective and constrained optimization.

TABLE III
COMPARISON OF HCLS-DDEA WITH OTHER ALGORITHMS

D	Fun	HCLS-DDEA	BDDEA-LDG	DDEA-PES	CC-DDEA	ESCO
10	F1	**9.84E-01 ± 6.85E-01**	1.07E+00 ± 4.03E-01 (≈)	1.34E+00 ± 6.20E-01 (+)	2.25E+00 ± 9.31E-01 (+)	9.53E+00 ± 5.74E+00 (+)
	F2	**2.53E+01 ± 3.77E+00**	3.35E+01 ± 8.28E+00 (+)	2.94E+01 ± 7.52E+00 (≈)	6.22E+01 ± 9.19E+00 (+)	4.27E+01 ± 1.02E+01 (+)
	F3	**5.89E+00 ± 7.52E-01**	6.65E+00 ± 6.13E-01 (+)	6.40E+00 ± 1.44E+00 (≈)	8.59E+00 ± 9.15E-01 (+)	8.53E+00 ± 1.75E+00 (+)
	F4	1.25E+00 ± 1.22E-01	1.27E+00 ± 1.26E-01 (≈)	1.31E+00 ± 1.36E-01 (≈)	**9.78E-01 ± 8.94E-02** (−)	5.31E+00 ± 2.00E+00 (+)
	F5	6.15E+01 ± 1.85E+01	6.62E+01 ± 2.85E+01 (≈)	5.99E+01 ± 1.76E+01 (≈)	9.65E+01 ± 2.72E+01 (+)	**4.52E+01 ± 7.74E+00** (−)
	+/≈ /−	NA	2/3/0	1/4/0	4/0/1	4/0/1
30	F1	**4.38E+00 ± 1.07E+00**	6.77E+00 ± 2.27E+00 (+)	5.76E+00 ± 2.27E+00 (+)	1.08E+01 ± 5.30E+00 (+)	2.22E+01 ± 5.39E+00 (+)
	F2	**5.70E+01 ± 4.08E+00**	6.94E+01 ± 7.79E+00 (+)	7.01E+01 ± 8.15E+00 (+)	1.46E+02 ± 2.92E+01 (+)	7.62E+01 ± 1.16E+01 (+)
	F3	**4.46E+00 ± 5.34E-01**	5.64E+00 ± 6.52E-01 (+)	5.45E+00 ± 3.63E-01 (+)	7.02E+00 ± 1.07E+00 (+)	5.87E+00 ± 4.21E-01 (+)
	F4	1.27E+00 ± 1.15E-01	1.38E+00 ± 1.09E-01 (+)	1.25E+00 ± 9.28E-02 (≈)	**1.00E+00 ± 5.26E-02** (−)	5.29E+00 ± 8.77E-01 (+)
	F5	**1.02E+02 ± 2.15E+01**	1.56E+02 ± 4.28E+01 (+)	1.48E+02 ± 2.99E+01 (+)	2.36E+02 ± 4.63E+01 (+)	1.20E+02 ± 2.13E+01 (+)
	+/≈ /−	NA	5/0/0	4/1/0	4/0/1	5/0/0
50	F1	1.49E+01 ± 4.67E+00	**1.31E+01 ± 3.04E+00** (≈)	1.43E+01 ± 4.57E+00 (≈)	2.42E+01 ± 7.14E+00 (+)	1.04E+02 ± 2.32E+01 (+)
	F2	**8.27E+01 ± 5.32E+00**	9.89E+01 ± 9.67E+00 (+)	1.09E+02 ± 1.06E+01 (+)	2.15E+02 ± 3.00E+01 (+)	3.57E+02 ± 7.98E+01 (+)
	F3	4.99E+00 ± 3.88E-01	**4.81E+00 ± 3.48E-01** (≈)	4.94E+00 ± 4.20E-01 (≈)	6.59E+00 ± 6.40E-01 (+)	8.13E+00 ± 7.14E-01 (+)
	F4	1.96E+00 ± 3.07E-01	1.41E+00 ± 8.79E-02 (−)	1.33E+00 ± 9.22E-02 (−)	**9.99E-01 ± 7.39E-02** (−)	1.42E+01 ± 4.14E+00 (+)
	F5	**1.84E+02 ± 2.70E+01**	1.96E+02 ± 3.04E+01 (≈)	2.35E+02 ± 4.84E+01 (+)	3.33E+02 ± 6.32E+01 (+)	1.95E+02 ± 2.39E+01 (≈)
	+/≈ /−	NA	1/3/1	2/2/1	4/0/1	4/1/0

TABLE IV
ABLATION STUDY RESULTS

Fun	*D* = 10		*D* = 30	
	HCLS-DDEA	Global-Only	HCLS-DDEA	Global-Only
F1	**9.84E-01 ± 6.85E-01**	1.11E+00 ± 3.72E-01 (≈)	**4.38E+00 ± 1.07E+00**	4.55E+00 ± 1.88E+00 (≈)
F2	**2.53E+01 ± 3.77E+00**	2.94E+01 ± 5.91E+00 (+)	**5.70E+01 ± 4.08E+00**	5.76E+01 ± 4.80E+00 (≈)
F3	5.89E+00 ± 7.52E-01	**5.60E+00 ± 1.15E+00 (≈)**	**4.46E+00 ± 5.34E-01**	5.01E+00 ± 3.80E-01 (+)
F4	**1.25E+00 ± 1.22E-01**	1.35E+00 ± 1.46E-01 (+)	1.27E+00 ± 1.15E-01	**1.23E+00 ± 9.35E-02 (≈)**
F5	6.15E+01 ± 1.85E+01	**5.45E+01 ± 2.17E+01 (≈)**	**1.02E+02 ± 2.15E+01**	1.12E+02 ± 2.86E+01 (≈)
+/≈ /−	NA	2/3/0	NA	1/4/0

REFERENCES

[1] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011.

[2] T. Chugh, K. Sindhya, J. Hakanen, and K. Miettinen, "A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms," *Soft Computing*, vol. 23, no. 9, pp. 3137–3166, 2019.

[3] Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, and K. Y. Lum, "Combining global and local surrogate models to accelerate evolutionary optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 1, pp. 66–76, 2006.

[4] M. D. Buhmann, "Radial basis functions," *Acta numerica*, vol. 9, pp. 1–38, 2000.

[5] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker, "Surrogate-based analysis and optimization," *Progress in aerospace sciences*, vol. 41, no. 1, pp. 1–28, 2005.

[6] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[7] Y. Jin, H. Wang, T. Chugh, D. Guo, and K. Miettinen, "Data-driven evolutionary optimization: An overview and case studies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, pp. 442–458, 2018.

[8] M. T. Emmerich, K. C. Giannakoglou, and B. Naujoks, "Single-and multiobjective evolutionary optimization assisted by gaussian random field metamodels," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 4, pp. 421–439, 2006.

[9] T. J. Santner, B. J. Williams, W. I. Notz, and B. J. Williams, *The design and analysis of computer experiments*. Springer, 2003, vol. 1.

[10] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[11] H. Wang, Y. Jin, C. Sun, and J. Doherty, "Offline data-driven evolutionary optimization using selective surrogate ensembles," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 2, pp. 203–216, 2018.

[12] J.-Y. Li, Z.-H. Zhan, C. Wang, H. Jin, and J. Zhang, "Boosting data-driven evolutionary algorithm with localized data generation," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 5, pp. 923–937, 2020.

[13] J. MacQueen, "Multivariate observations," in *Proceedings ofthe 5th Berkeley Symposium on Mathematical Statisticsand Probability*, vol. 1, 1967, pp. 281–297.

[14] C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng, "Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 4, pp. 644–660, 2017.

[15] J.-Y. Li, Z.-H. Zhan, H. Wang, and J. Zhang, "Data-driven evolutionary algorithm with perturbation-based ensemble surrogates," *IEEE transactions on cybernetics*, vol. 51, no. 8, pp. 3925–3937, 2020.

[16] Y.-J. Gong, Y.-T. Zhong, and H.-G. Huang, "Offline data-driven optimization at scale: A cooperative coevolutionary approach," *IEEE Transactions on Evolutionary Computation*, vol. 28, no. 6, pp. 1809–1823, 2023.

[17] X. Wu, Q. Lin, J. Li, K. C. Tan, and V. C. Leung, "An ensemble surrogate-based coevolutionary algorithm for solving large-scale expensive optimization problems," *IEEE Transactions on Cybernetics*, vol. 53, no. 9, pp. 5854–5866, 2022.