Multiform Surrogate-Assisted Evolutionary Algorithm for Expensive Optimization Problems with Mixed Variables

Shenghao Wu
South China Agricultural University
Guangzhou, China
wushenghao8404@qq.com

Xiaoming Xue
China University of Petroleum (East China)
Qingdao, China
xuexm@upc.edu.cn

***Abstract*—Expensive optimization problems with mixed continuous and categorical variables (EOPMVs) arise in engineering and scientific design, yet remain difficult because evaluations are costly and the two variable types interact in non-smooth ways. To address this challenge, we propose multiform transfer ant colony optimization (MFT-ACO), a surrogate-assisted framework grounded in the multiform transfer optimization paradigm. MFT-ACO reformulates the target problem into three complementary auxiliary formulations by exploiting the distinct modeling strengths of different surrogates for heterogeneous variables: an RBF-based full-space formulation for smooth interpolation, an LSBT-based combinatorial formulation for categorical discontinuities, and a category-conditioned local refinement formulation for continuous improvement in promising regions. These forms co-evolve through an iterative multiform selection mechanism that enables additional surrogate-guided refinement before expensive evaluations are performed. Experimental results on benchmark EOPMVs show that MFT-ACO is competitive under a limited evaluation budget, suggesting that multiform reformulation is a viable strategy for expensive mixed-variable optimization.**

***Index Terms*—Surrogate modeling, mixed-variable optimization, ant colony optimization.**

I. Introduction

Many real-world engineering design tasks involve optimizing systems with both continuous variables (e.g., dimensions, temperatures) and categorical variables (e.g., material types, component configurations). When the objective function requires computationally intensive simulations, each evaluation becomes costly. Such tasks are commonly referred to as expensive optimization problems with mixed variables (EOPMVs) [1]–[4]. Formally, an EOPMV can be defined as:

$$\min_{\mathbf{x}^{\text{con}},\mathbf{x}^{\text{cat}}} f(\mathbf{x}^{\text{con}},\mathbf{x}^{\text{cat}}), \tag{1}$$

subject to $\mathbf{L}^{\text{con}} \le \mathbf{x}^{\text{con}} \le \mathbf{U}^{\text{con}}$, $\mathbf{x}^{\text{cat}}_j \in \mathcal{V}_j$ for $j = 1,\ldots,n_2$, where $\mathbf{x}^{\text{con}} \in \mathbb{R}^{n_1}$ and $\mathbf{x}^{\text{cat}} \in \prod_{j=1}^{n_2} \mathcal{V}_j$ denote the continuous and categorical variables, respectively. The two variable types often interact non-additively, making the landscape non-smooth and difficult to model.

Surrogate-assisted evolutionary algorithms (SAEAs) [5]–[10] address this bottleneck by combining the global search capability of evolutionary computation [11], [12] with fast surrogate models that replace expensive evaluations. However, most SAEAs target purely continuous problems [13]–[15]. Smooth surrogates like radial basis functions (RBFs) [16] distort the landscape when applied to mixed-variable problems, while tree-based models like least-squares boosting trees (LSBTs) [17] handle categorical variables naturally but lack the smoothness needed for continuous refinement.

Motivated by this, we propose a multiform transfer optimization (MFTO) framework that treats different surrogate-induced problem formulations as complementary ***forms*** of the same target problem. We define three forms: (1) an RBF-based full-space formulation for smooth interpolation; (2) an LSBT-based combinatorial formulation for categorical discontinuities; and (3) a category-conditioned local refinement formulation that builds a dedicated RBF on the continuous subspace conditioned on the best categorical configuration. These forms co-evolve through iterative surrogate-guided refinement before expensive evaluations.

The main contributions of this work are:

1) A multiform co-evolution mechanism that iteratively refines candidates under complementary RBF and LSBT surrogates over multiple generations.
2) A category-conditioned local refinement formulation that constructs a context-specific RBF on the continuous subspace, enabling gradient-based refinement via an interior-point method.
3) A unified framework combining global exploration across heterogeneous surrogates with focused local exploitation for EOPMVs under tight evaluation budgets.

II. Preliminaries and Related Work

Our approach integrates three components: the RBF and LSBT surrogates as complementary models, the mixed-variable ant colony optimization (ACO_{MV}) for structured search, and the MFTO paradigm for co-evolutionary coordination.

Corresponding author: Xiaoming Xue (xuexm@upc.edu.cn).

A. Surrogate Models for Mixed-Variable Problems

The RBF surrogate is a kernel-based interpolant that models the objective as $f_{\text{RBF}}(\mathbf{x}) = \sum_{k=1}^{N} w_k\, \kappa(\delta(\mathbf{x}, \mathbf{x}_k))$, where $\kappa(\cdot)$ is a Gaussian kernel. For mixed-variable problems, we adopt a hybrid distance combining Euclidean and Hamming components [1]:

$$\delta\big((\mathbf{x}^{\text{con}}, \mathbf{x}^{\text{cat}}), (\mathbf{x}_k^{\text{con}}, \mathbf{x}_k^{\text{cat}})\big) = \sqrt{\|\mathbf{x}^{\text{con}} - \mathbf{x}_k^{\text{con}}\|_2^2 + \rho(\mathbf{x}^{\text{cat}}, \mathbf{x}_k^{\text{cat}})}, \tag{2}$$

where ρ counts mismatched categorical attributes. This allows the RBF to respect categorical discontinuities while maintaining smooth interpolation in the continuous subspace.

The LSBT surrogate [17] is an ensemble of shallow regression trees trained via gradient boosting: $f_{\text{LSBT}}(\mathbf{x}) = \sum_{b=1}^{B} T(\mathbf{x}; \Theta_b)$, where each $T(\cdot; \Theta_b)$ is a regression tree and B is the number of boosting rounds. LSBT natively supports categorical inputs through explicit splits without numerical encoding, and its piecewise-constant structure suits non-smooth or discrete-dominant landscapes where RBF may generalize poorly.

B. Ant Colony Optimization for Mixed-Variable Problems

A well-established optimizer for mixed-variable problems is ACO_{MV} [18], which extends the continuous ant colony optimization framework to handle both continuous and categorical variables. The algorithm maintains a solution archive ($\texttt{SA}$) of size K, storing the K best evaluated solutions in ascending order of objective value. Each solution in the archive is denoted as $\mathbf{s}_k = [\mathbf{s}_k^{\text{con}}, \mathbf{s}_k^{\text{cat}}]$, where $\mathbf{s}_k^{\text{con}} \in \mathbb{R}^{n_1}$ represents the continuous part and $\mathbf{s}_k^{\text{cat}} \in \prod_{j=1}^{n_2} \mathcal{V}_j$ denotes the categorical part. New candidate solutions are generated through two coordinated mechanisms:

(1) Continuous variables are sampled from a Gaussian distribution centered at the continuous component of a selected parent solution. Specifically, let $\mathbf{s}^{\text{con}} = [s_1^{\text{con}}, \ldots, s_{n_1}^{\text{con}}]$ be the continuous part of a chosen parent. The i-th continuous variable of an offspring is sampled as

$$x_i^{\text{con}} \sim \mathcal{N}\big(s_i^{\text{con}}, \sigma_i^2\big), \tag{3}$$

where the standard deviation σ_i is computed as

$$\sigma_i = \xi \cdot \frac{1}{K-1} \sum_{r=1}^{K} \left|\texttt{SA}(r)_i^{\text{con}} - s_i^{\text{con}}\right|, \tag{4}$$

and $\xi > 0$ is a search-width parameter that controls exploration intensity.

(2) Categorical variables are generated via weighted sampling over each category's candidate set $\mathcal{V}_j = \{v_1, \ldots, v_{l_j}\}$. The selection weight for value v_t in variable j is defined as:

$$\beta_t = \begin{cases} \dfrac{\alpha_j^t}{u_j^t} + \dfrac{q}{\eta_j}, & \text{if } \eta_j > 0 \text{ and } u_j^t > 0, \\ \dfrac{q}{\eta_j}, & \text{if } \eta_j > 0 \text{ and } u_j^t = 0, \\ \dfrac{\alpha_j^t}{u_j^t}, & \text{if } \eta_j = 0 \text{ and } u_j^t > 0, \end{cases} \tag{5}$$

Algorithm 1 $\text{ACO}_{\text{MV}}(\texttt{SA})$

Require: solution archive $\texttt{SA}$ of size K; parameters $\xi > 0$, $q > 0$, offspring size M
Ensure: offspring pool $\texttt{OP}$ of size M

Initialize $\texttt{OP} \leftarrow$ empty list;
for $i = 1$ to M **do**
  Compute weights ω_k $(k = 1, \ldots, K)$ using Eq. (6);
  Sample a parent index k with probability in Eq. (7);
  for each continuous dimension $d = 1$ to n_1 **do**
    Compute σ_d using Eq. (4);
    Sample $x_d^{\text{con}} \sim \mathcal{N}(\texttt{SA}(k)_d^{\text{con}}, \sigma_d^2)$;
  end for
  for each categorical variable $j = 1$ to n_2 **do**
    Compute weights β_t for all $v_t \in \mathcal{V}_j$ using Eq. (5);
    Sample $x_j^{\text{cat}} \in \mathcal{V}_j$ based on normalized β_t;
  end for
  Append $\mathbf{x}_i = [\mathbf{x}^{\text{con}}, \mathbf{x}^{\text{cat}}]$ to $\texttt{OP}$;
end for
return $\texttt{OP}$

where u_j^t is the number of solutions in $\texttt{SA}$ that use category value v_t for variable j, α_j^t is the highest pheromone weight among those solutions, η_j is the number of unused categories in $\mathcal{V}_j$, and $q > 0$ is a parameter that promotes exploration of unseen categories.

The pheromone weight associated with the k-th solution in $\texttt{SA}$ is rank-based:

$$\omega_k = \frac{1}{qK\sqrt{2\pi}} \exp\left(-\frac{(r_k - 1)^2}{2q^2K^2}\right), \tag{6}$$

where $r_k \in \{1, \ldots, K\}$ is the rank of solution k (with rank 1 assigned to the best solution). A parent solution is selected with probability proportional to its pheromone weight:

$$p_k = \frac{\omega_k}{\sum_{r=1}^{K} \omega_r}. \tag{7}$$

This dual-generation strategy enables ACO_{MV} to balance exploitation of high-quality regions and exploration of underrepresented categorical configurations. In the proposed framework, ACO_{MV} is used only for offspring generation, while initialization, expensive evaluation, and archive update are handled by the outer algorithm, as summarized in **Algorithm 1**.

C. Multiform Transfer Optimization

MFTO [19], [20] is a paradigm within evolutionary transfer optimization [21]–[24] that tackles a complex problem by constructing alternative formulations from the original problem. Different forms expose different structural aspects: some favor global exploration, others local refinement. Knowledge transfer between forms can be implicit (shared populations) or explicit (solution migration). The paradigm has been applied to large-scale [25], expensive [26], and constrained optimization [27], demonstrating that multiform reformulation can improve search efficiency by leveraging representational flexibility of complex problems.

Algorithm 2 MFT-ACO Framework

Require: problem instance, maximum function evaluations (`MaxFEs`)
Ensure: best solution found
Initialize `SA` via Latin hypercube sampling;
Evaluate `SA` and store in `DB`;
`FEs` $\leftarrow$ size(`DB`);
while `FEs` $<$ `MaxFEs` **do**
  `OP` $\leftarrow$ ACO$_{\text{MV}}$(`SA`); ▷ **Algorithm 1**
  $\mathtt{X}_{\text{sel}} \leftarrow$ MULTIFORMITERATIVESELECTION(`OP`, `DB`) ; ▷ **Algorithm 3**
  Evaluate $\mathtt{X}_{\text{sel}}$;
  Update `DB` and `FEs`;
  Update `SA` via elitist replacement;
end while
return best solution in `DB`

III. MULTIFORM TRANSFER ANT COLONY OPTIMIZATION FRAMEWORK

A. Framework Overview

As discussed in [1], an RBF with Hamming-augmented distance can capture smooth trends in the mixed space but may become unreliable when categorical effects dominate, while LSBT handles categorical discontinuities naturally but lacks smoothness for continuous refinement. This mismatch motivates an MFTO view: different surrogate-induced formulations serve as complementary forms, providing both *complementarity* and *controlled transfer*. Accordingly, we propose MFT-ACO, a surrogate-assisted evolutionary algorithm for EOPMVs that instantiates three such formulations as interacting auxiliary problems. As shown in **Algorithm 2**, MFT-ACO maintains: (1) a solution archive (`SA`) for offspring generation via ACO$_{\text{MV}}$; (2) a database (`DB`) of all evaluated solutions for surrogate construction; and (3) a MULTIFORMITERATIVESELECTION routine (Section III-B) with two form-specific subpopulations $\mathbf{P}_1$ (RBF) and $\mathbf{P}_2$ (LSBT). At each iteration, offspring generated by ACO$_{\text{MV}}$ undergo iterative co-evolution under both surrogates, selected candidates from all three forms are evaluated on the true objective, and `SA` is updated via elitist replacement.

B. Multiform Iterative Selection

The MULTIFORMITERATIVESELECTION routine coordinates the co-evolution of the three forms. **Form 1** is an RBF-based full-space formulation with Hamming-augmented distance, intended to provide a smooth approximation over the mixed domain. **Form 2** is an LSBT-based combinatorial formulation, intended to represent categorical discontinuities without distance-based encoding. **Form 3** is a category-conditioned local refinement formulation that is activated only when enough evaluated solutions share the current best categorical configuration. In that case, a dedicated RBF is constructed on the associated continuous subspace and refined by an interior-point method. From the MFTO perspective, this third form can be viewed as a variable-decoupled auxiliary formulation induced by conditioning on the categorical assignment.

As shown in **Algorithm 3**, two subpopulations, $\mathbf{P}_1$ and $\mathbf{P}_2$, are initialized from the top-N_p offspring according to the RBF and LSBT predictions, respectively. Over T inner iterations, each subpopulation serves as the archive for ACO$_{\text{MV}}$ to generate new candidates. The offspring produced from the two forms are pooled, inserted into both subpopulations, and rescored by the corresponding surrogate before truncation to size N_p. This shared-offspring mechanism implements implicit transfer between formulations: candidates favored under one formulation remain available for selection under the other, while repeated truncation preserves selection pressure and limits uncontrolled growth. After T iterations, elite candidates not already contained in `DB` are extracted as $\mathcal{K}_1$ from $\mathbf{P}_1$ and $\mathcal{K}_2$ from $\mathbf{P}_2$. When the conditioning data are sufficient, Form 3 yields $\mathcal{K}_3$ through category-conditioned local refinement. The final set $\mathtt{X}_{\text{sel}} = \mathcal{K}_1 \cup \mathcal{K}_2 \cup \mathcal{K}_3$ therefore aggregates candidates from all active formulations for expensive evaluation. In this way, true evaluations are deferred until after several rounds of surrogate-guided refinement, which reduces premature commitment to any single surrogate and is intended to improve data efficiency under tight evaluation budgets.

IV. EXPERIMENTAL STUDIES

A. Experimental Setup

We adopt the same set of 30 artificial EOPMVs as used in [1], labeled F1 through F30. These benchmark problems are constructed by embedding categorical decision variables into five classical continuous test functions: Sphere, Rosenbrock, Ackley, Ellipsoid, and Griewank. Each problem is defined over a mixed-variable search space consisting of n_1 continuous variables, each bounded within $[-100, 100]$, and n_2 categorical variables, where each categorical variable takes values from a discrete set of size 3, 5, or 10. All problems possess known global optima with an objective value of zero, enabling precise performance assessment.

Following the classification in [1], the 30 benchmarks are grouped into three types based on the relative proportion of continuous and categorical variables. Type 1 includes F1–F10, where the majority of variables are continuous ($n_1 = 8$, $n_2 = 2$). Type 2 comprises F11–F20, featuring a balanced mix of continuous and categorical variables ($n_1 = 5$, $n_2 = 5$). Type 3 consists of F21–F30, in which categorical variables dominate ($n_1 = 2$, $n_2 = 8$). This systematic design ensures a comprehensive evaluation across diverse landscape characteristics, including separability, multimodality, ill-conditioning, and strong interactions between continuous and categorical dimensions.

We compare MFT-ACO against five representative SAEAs designed for expensive optimization problems:

- **SACOSO** [13]: A cooperative swarm optimization algorithm that partitions the search space and employs multiple surrogates to handle high-dimensional expensive problems.
- **SADE-AMSS** [28]: A differential evolution variant that dynamically allocates computational resources to promising subspaces through adaptive multi-subspace search.
- **SADE-ATDSC** [29]: Enhances SADE by adaptively selecting training data based on a dynamic criterion to improve surrogate fidelity during evolution.

Algorithm 3 MultiformIterativeSelection($\texttt{OP}$, $\texttt{DB}$)

Require: offspring pool $\texttt{OP}$, database $\texttt{DB}$, co-evolution iterations T, subpopulation size N_p, local-search threshold $N_{\min}$
Ensure: selected solutions $\texttt{X}_{\text{sel}}$

Construct RBF surrogate f_{RBF} using Hamming-augmented distance on $\texttt{DB}$;
Construct LSBT surrogate f_{LSBT} on $\texttt{DB}$ with categorical predictors specified;
Initialize $\mathbf{P}_1$.decs $\leftarrow \texttt{OP}$ ▷ Form 1: RBF-guided;
Initialize $\mathbf{P}_2$.decs $\leftarrow \texttt{OP}$ ▷ Form 2: LSBT-guided;
Evaluate $\mathbf{P}_1$.objs $\leftarrow f_{\text{RBF}}(\mathbf{P}_1$.decs);
Evaluate $\mathbf{P}_2$.objs $\leftarrow f_{\text{LSBT}}(\mathbf{P}_2$.decs);
Truncate $\mathbf{P}_1$ to top-N_p individuals;
Truncate $\mathbf{P}_2$ to top-N_p individuals;
for $t = 1$ to T **do**
    $\texttt{Off}_1 \leftarrow \text{ACO}_{\text{MV}}(\mathbf{P}_1$.decs); ▷ **Algorithm 1**
    $\texttt{Off}_2 \leftarrow \text{ACO}_{\text{MV}}(\mathbf{P}_2$.decs); ▷ **Algorithm 1**
    $\mathbf{P}_1$.decs $\leftarrow [\mathbf{P}_1$.decs$; \texttt{Off}_1; \texttt{Off}_2]$;
    $\mathbf{P}_2$.decs $\leftarrow [\mathbf{P}_2$.decs$; \texttt{Off}_1; \texttt{Off}_2]$;
    Re-evaluate $\mathbf{P}_1$.objs $\leftarrow f_{\text{RBF}}(\mathbf{P}_1$.decs);
    Re-evaluate $\mathbf{P}_2$.objs $\leftarrow f_{\text{LSBT}}(\mathbf{P}_2$.decs);
    Truncate $\mathbf{P}_1$ to top-N_p individuals;
    Truncate $\mathbf{P}_2$ to top-N_p individuals;
end for
// Extract knowledge from each form
Identify elite candidates $\mathcal{K}_1$ from $\mathbf{P}_1$ not in $\texttt{DB}$; ▷ Knowledge from Form 1
Identify elite candidates $\mathcal{K}_2$ from $\mathbf{P}_2$ not in $\texttt{DB}$; ▷ Knowledge from Form 2
// Form 3: Category-conditioned local refinement
Find current best solution $\mathbf{x}^* = [\mathbf{x}^{*\text{con}}, \mathbf{x}^{*\text{cat}}]$ in $\texttt{DB}$;
Identify all solutions in $\texttt{DB}$ with same categorical part as $\mathbf{x}^{*\text{cat}}$;
Let N_{ls} be the number of such solutions;
if $N_{\text{ls}} > N_{\min}$ **then**
    Extract their continuous parts $\{\mathbf{x}_i^{\text{con}}\}$ and objective values $\{f_i\}$;
    Build local RBF surrogate f_{local} on $\{\mathbf{x}_i^{\text{con}}, f_i\}$;
    Use an interior-point method to solve $\min_{\mathbf{x}^{\text{con}}} f_{\text{local}}(\mathbf{x}^{\text{con}})$ s.t. bounds;
    Let $\hat{\mathbf{x}}^{\text{con}}$ be the optimized continuous part;
    Form candidate $\hat{\mathbf{x}} = [\hat{\mathbf{x}}^{\text{con}}, \mathbf{x}^{*\text{cat}}]$;
    if $\hat{\mathbf{x}} \notin \texttt{DB}$ **then**
        $\mathcal{K}_3 \leftarrow \{\hat{\mathbf{x}}\}$ ▷ Knowledge from Form 3;
    else
        $\mathcal{K}_3 \leftarrow \emptyset$;
    end if
else
    $\mathcal{K}_3 \leftarrow \emptyset$;
end if
// Fuse knowledge from all three forms
$\texttt{X}_{\text{sel}} \leftarrow \mathcal{K}_1 \cup \mathcal{K}_2 \cup \mathcal{K}_3$;
return $\texttt{X}_{\text{sel}}$

- **SADE-Sammon** [30]: Integrates Sammon mapping for dimensionality reduction, enabling efficient surrogate modeling in a reduced space for high-dimensional problems.
- **SAMSO** [14]: A multi-swarm optimization framework that maintains diverse subpopulations guided by ensemble surrogates to balance exploration and exploitation.

All compared algorithms are configured consistently with the settings reported in [1]. The maximum number of function evaluations is fixed at 600 for all methods. The solution archive ($\texttt{SA}$) size is set to 60, and the offspring population generated per iteration contains 100 candidate solutions. The initial population is created using Latin hypercube sampling to ensure uniform coverage of the search space. For the surrogate-assisted local refinement component, the activation threshold is defined as $N_{\min} = 5 \cdot n_2$, meaning that local refinement is triggered only when at least $5 \cdot n_2$ evaluated solutions share the same categorical configuration as the current best solution.

For the proposed MFT-ACO algorithm, the co-evolution loop in the multiform selection phase runs for $T = 10$ iterations. Each form-specific subpopulation maintains a size of $N_p = 60$ individuals. The RBF surrogate employs a Gaussian kernel with Hamming-augmented distance to handle mixed variables, while the LSBT surrogate is implemented using MATLAB's `fitrensemble` function with default boosting parameters and explicit specification of categorical predictors. To account for stochastic variation, each algorithm is executed independently 30 times on every test problem. All implementations are integrated into the PlatEMO platform to ensure consistent evaluation protocols, termination conditions, and data management.

The performance of all algorithms is evaluated using the best-so-far objective value at the 600th function evaluation. Results are reported as mean $\pm$ standard deviation over 30 independent runs. Statistical significance is assessed via the Wilcoxon rank-sum test at the 5% significance level, with symbols $+$, $-$, and $=$ indicating that the competitor is significantly better than, worse than, or equivalent to MFT-ACO, respectively.

B. Results and Analysis

Table I summarizes the results on 30 EOPMV benchmarks. MFT-ACO attains the lowest mean objective value on 26 of 30 problems and is significantly better than SACOSO, SADE-Sammon, and SAMSO on all 30 instances (0/30/0).

On Type 1 problems (F1–F10, continuous-dominant), MFT-ACO achieves the best mean on 9 of 10 instances, reducing objective values by one to four orders of magnitude over most competitors. The exception is F6, where SADE-ATDSC achieves a notably lower mean (4.68 vs. 20.89) through adaptive training-data selection. On Type 2 problems (F11–F20, balanced mix), MFT-ACO ranks first on 8 of 10; the two losses against SADE-AMSS occur on F16 and F19, both Sphere-based instances where SADE-AMSS's adaptive subspace search is particularly effective on separable landscapes. On Type 3 problems (F21–F30, categorical-dominant), MFT-ACO achieves top performance on 9 of 10; on the exception F22 (Rosenbrock, $n_2 = 8$), SADE-ATDSC reports a lower mean but with no statistical significance. Overall, MFT-ACO wins on 23, ties on 5, and loses on 2 against SADE-AMSS, and wins on 19, ties on 10, and loses on 1 against SADE-ATDSC, confirming that multiform co-evolution adapts consistently across different proportions of variable types.

C. Ablation Studies

To validate each component, we compare MFT-ACO against three ablated variants: MFT-ACO-w/o-RBF (no RBF form), MFT-ACO-w/o-LSBT (no LSBT form), and MFT-ACO-w/o-LS (no local refinement form). Table II reports the detailed results.

Effect of RBF (Form 1). Removing the RBF form yields a record of $4/14/12$. On Type 1 problems, where continuous variables dominate, the absence of smooth interpolation

TABLE I

COMPARISON OF BEST-SO-FAR OBJECTIVE VALUES AT 600 FUNCTION EVALUATIONS ON 30 EOPMV BENCHMARK PROBLEMS (MEAN $\pm$ STD OVER 30 RUNS). BEST MEANS ARE HIGHLIGHTED IN **BOLD**.

Problem	SACOSO	SADE-AMSS	SADE-ATDSC	SADE-Sammon	SAMSO	MFT-ACO
F1	1.7142e+3 (8.94e+2) −	2.7397e+2 (1.81e+2) −	1.8180e+2 (1.36e+2) −	2.6124e+3 (8.62e+2) −	4.9165e+2 (4.72e+2) −	**5.5904e+0 (2.12e+1)**
F2	7.5528e+1 (8.59e+0) −	6.7701e+1 (1.20e+1) −	3.0269e+1 (1.25e+1) =	6.7120e+1 (1.49e+1) −	7.2852e+1 (2.05e+1) −	**2.5356e+1 (9.48e+0)**
F3	1.3406e+1 (1.33e+0) −	5.6209e+0 (1.45e+0) −	4.4783e+0 (1.82e+0) −	1.0915e+1 (1.19e+0) −	1.7054e+1 (1.27e+0) −	**2.0025e−4 (1.50e−4)**
F4	7.1120e+1 (2.25e+1) −	8.0756e+0 (1.02e+1) −	4.4483e+0 (7.83e+0) =	4.8845e+1 (1.74e+1) −	1.0793e+1 (1.56e+1) −	**1.0408e−4 (1.92e−4)**
F5	4.0482e+1 (1.31e+1) −	3.7315e+0 (1.62e+0) −	7.1289e−1 (1.70e+0) =	2.7235e+1 (7.04e+0) −	1.9962e+1 (1.35e+1) −	**5.1426e−1 (3.04e−1)**
F6	3.9990e+3 (1.06e+3) −	2.1357e+2 (2.59e+2) −	**4.6770e+0 (1.68e+1)** +	3.3408e+3 (1.18e+3) −	7.4441e+2 (1.54e+3) −	2.0888e+1 (3.38e+1)
F7	8.0020e+1 (1.15e+1) −	6.6683e+1 (1.37e+1) −	3.6561e+1 (1.81e+1) =	7.5119e+1 (9.93e+0) −	6.6855e+1 (1.85e+1) −	**3.0771e+1 (1.20e+1)**
F8	1.3745e+1 (1.19e+0) −	5.6589e+0 (2.43e+0) −	3.1109e+0 (3.50e+0) =	1.1115e+1 (1.33e+0) −	1.6927e+1 (1.32e+0) −	**2.0059e−1 (6.17e−1)**
F9	3.7099e+1 (2.02e+1) −	7.3171e+0 (1.06e+1) −	2.3564e+0 (8.35e+0) =	3.1617e+1 (1.05e+1) −	1.5627e+1 (1.21e+1) −	**9.2969e−2 (1.76e−1)**
F10	3.8257e+1 (1.24e+1) −	5.3868e+0 (4.65e+0) −	4.4861e+0 (1.26e+1) −	3.3380e+1 (1.28e+1) −	2.7043e+1 (1.47e+1) −	**7.9820e−1 (3.91e−1)**
F11	1.5887e+3 (1.87e+3) −	3.4813e+1 (4.73e+1) =	4.4038e+2 (6.08e+2) −	1.0224e+3 (4.37e+2) −	1.4368e+3 (1.30e+3) −	**2.2128e+1 (2.87e+1)**
F12	7.6976e+1 (2.38e+1) −	5.9734e+1 (2.28e+1) −	5.6492e+1 (1.15e+1) −	6.2426e+1 (1.81e+1) −	6.9584e+1 (1.84e+1) −	**3.9683e+1 (1.58e+1)**
F13	1.0504e+1 (2.74e+0) −	3.1346e+0 (1.31e+0) −	6.4883e+0 (2.71e+0) −	9.7525e+0 (2.40e+0) −	9.6102e+0 (2.51e+0) −	**1.5739e−3 (7.03e−3)**
F14	7.7882e+0 (1.53e+1) −	4.5981e−1 (3.62e−1) =	4.5743e+0 (7.86e+0) −	1.2137e+1 (8.86e+0) −	1.5627e+1 (2.07e+1) −	**3.4900e−1 (2.95e−1)**
F15	2.4579e+1 (1.28e+1) −	2.8945e+0 (2.31e+0) −	4.0829e+0 (4.68e+0) =	7.6562e+0 (5.23e+0) −	7.7167e+0 (1.07e+1) −	**1.8773e+0 (2.80e+0)**
F16	1.3332e+3 (1.08e+3) −	**1.6556e+2 (2.63e+2)** +	1.1678e+3 (1.10e+3) −	2.1547e+3 (1.79e+3) −	2.0142e+3 (1.86e+3) −	2.9424e+2 (3.77e+2)
F17	1.0011e+2 (1.60e+1) −	6.4139e+1 (1.39e+1) −	7.2278e+1 (1.46e+1) −	7.6420e+1 (1.24e+1) −	7.3941e+1 (1.72e+1) −	**4.6566e+1 (1.05e+1)**
F18	7.6220e+0 (3.75e+0) −	4.1914e+0 (2.27e+0) −	7.1420e+0 (2.71e+0) −	1.0949e+1 (2.78e+0) −	1.1367e+1 (3.02e+0) −	**4.2332e−1 (1.32e+0)**
F19	1.5319e+2 (5.98e+1) −	**9.7834e−1 (8.48e−1)** +	1.4130e+1 (2.63e+1) −	1.7890e+1 (2.82e+1) −	3.0337e+1 (2.88e+1) −	1.6299e+0 (1.01e+0)
F20	4.4304e+1 (2.43e+1) −	3.5293e+0 (3.78e+0) =	1.6700e+1 (2.11e+1) −	2.3253e+1 (2.87e+1) −	1.8389e+1 (1.98e+1) −	**3.2344e+0 (1.86e+0)**
F21	2.4457e+3 (1.19e+3) −	2.6055e+2 (2.55e+2) −	6.3353e+2 (7.40e+2) −	2.3765e+3 (1.09e+3) −	1.4117e+3 (1.11e+3) −	**3.3086e+1 (1.89e+1)**
F22	9.8281e+1 (1.38e+1) −	6.3184e+1 (1.23e+1) −	**4.3885e+1 (1.13e+1)** =	7.2921e+1 (1.68e+1) −	7.7816e+1 (1.22e+1) −	5.0427e+1 (8.74e+0)
F23	9.7999e+0 (1.57e+0) −	4.6798e+0 (2.12e+0) −	5.3719e+0 (3.33e+0) −	1.1335e+1 (2.11e+0) −	1.3245e+1 (2.26e+0) −	**2.3573e−5 (4.52e−5)**
F24	5.9454e+1 (3.62e+1) −	6.1343e+0 (1.59e+1) −	3.5899e+0 (6.20e+0) =	3.2714e+1 (1.98e+1) −	2.1959e+1 (2.38e+1) −	**5.4407e−1 (5.60e−1)**
F25	5.0186e+1 (3.50e+1) −	3.8702e+0 (3.33e+0) −	3.4785e+0 (6.12e+0) =	2.9534e+1 (1.43e+1) −	3.7391e+1 (3.03e+1) −	**6.5720e−1 (4.51e−1)**
F26	4.6356e+3 (1.86e+3) −	2.5770e+2 (4.46e+2) =	5.4243e+2 (4.06e+2) −	3.0062e+3 (1.21e+3) −	1.7033e+3 (2.05e+3) −	**1.3699e+2 (6.77e+1)**
F27	1.0145e+2 (1.63e+1) −	7.0628e+1 (1.78e+1) −	6.8730e+1 (2.28e+1) −	8.2449e+1 (1.51e+1) −	7.8299e+1 (1.77e+1) −	**4.8160e+1 (1.18e+1)**
F28	1.1520e+1 (1.60e+0) −	7.0034e+0 (1.71e+0) −	6.4214e+0 (3.15e+0) −	1.1016e+1 (1.57e+0) −	1.4968e+1 (2.06e+0) −	**2.1516e−1 (5.28e−1)**
F29	6.0026e+1 (1.99e+1) −	5.8401e+0 (5.09e+0) =	8.4334e+0 (6.26e+0) −	2.7897e+1 (1.27e+1) −	4.6102e+1 (2.24e+1) −	**2.6647e+0 (1.25e+0)**
F30	5.4939e+1 (3.35e+1) −	5.9129e+0 (7.71e+0) −	6.2188e+0 (1.22e+1) −	4.6075e+1 (2.62e+1) −	5.8644e+1 (4.36e+1) −	**2.7721e+0 (1.51e+0)**
+/ − / =	0/30/0	2/23/5	1/19/10	0/30/0	0/30/0	/

leads to consistent degradation. However, on several problems across different types (F7, F15, F24, F30), the variant without RBF actually outperforms the full model, suggesting that RBF's distance-based interpolation can occasionally mislead the search when categorical effects dominate.

Effect of LSBT (Form 2). Removing the LSBT form causes the largest deterioration ($0/27/3$), particularly on Type 2 and Type 3 problems. On F11 (n_1=5, n_2=5), the mean increases from 2.21e+1 to 2.88e+3, confirming that the LSBT form is indispensable for handling categorical discontinuities.

Effect of local refinement (Form 3). Removing local refinement results in a record of $0/21/9$, with the largest impact on Ackley-based problems (F3, F8, F13, F23, F28), where the conditioned local RBF enables fine-grained continuous exploitation that the two global forms cannot provide.

V. CONCLUSION

We proposed MFT-ACO, a surrogate-assisted evolutionary algorithm for EOPMVs. The method couples an RBF-based full-space formulation, an LSBT-based combinatorial formulation, and a category-conditioned local refinement formulation within a multiform co-evolution framework. On 30 benchmark problems, MFT-ACO achieves the best mean performance on 26 of 30 problems and shows favorable Wilcoxon statistics against both baseline algorithms and ablated variants under a budget of 600 evaluations. These results suggest that multiform reformulation is a promising direction for expensive mixed-variable optimization, while the remaining hard cases indicate that adaptive coordination among forms deserves further study.

REFERENCES

[1] J. Liu, Y. Wang, G. Sun, and T. Pang, "Multisurrogate-assisted ant colony optimization for expensive optimization problems with continuous and categorical variables," *IEEE Transactions on Cybernetics*, vol. 52, no. 11, pp. 11 348–11 361, Nov. 2022.

[2] J.-Y. Li, Z.-H. Zhan, and J. Zhang, "Evolutionary computation for expensive optimization: A survey," *Machine Intelligence Research*, vol. 19, no. 1, pp. 3–23, Feb. 2022.

[3] B. J. Shields, J. Stevens, J. Li, M. Parasram, F. Damani, J. I. M. Alvarado, J. M. Janey, R. P. Adams, and A. G. Doyle, "Bayesian reaction optimization as a tool for chemical synthesis," *Nature*, vol. 590, no. 7844, pp. 89–96, Feb. 2021.

[4] S.-H. Wu, Z.-H. Zhan, K. C. Tan, and J. Zhang, "Traffic signal timing optimization: From evolution to adaptation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 55, no. 11, pp. 8341–8356, Nov. 2025.

[5] Z. Zhang, Y. Wang, G. Sun, T. Pang, and K. Tang, "Constrained probabilistic pareto dominance for expensive constrained multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 4, pp. 1138–1152, Aug. 2025.

[6] Z. Zhang, Y. Wang, G. Sun, and T. Pang, "A distribution information-based kriging-assisted evolutionary algorithm for expensive many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 6, pp. 2656–2670, Dec. 2025.

[7] Z. Zhang, Y. Wang, J. Liu, G. Sun, and K. Tang, "A two-phase kriging-assisted evolutionary algorithm for expensive constrained multiobjective optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 8, pp. 4579–4591, Aug. 2024.

[8] Z. Zhang, Y. Wang, G. Sun, and T. Pang, "A novel evolutionary bayesian optimization algorithm based on decomposition for expensive constrained multiobjective optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 56, no. 2, pp. 945–957, Feb. 2026.

[9] X. Wu, S. Wu, J. Wu, L. Feng, and K. C. Tan, "Evolutionary computation in the era of large language model: Survey and roadmap," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 2, pp. 534–554, Apr. 2025.

[10] S.-H. Wu, Z.-H. Zhan, and J. Zhang, "SAFE: Scale-adaptive fitness evaluation method for expensive optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 3, pp. 478–491, Jun. 2021.

[11] X. Huang, Z. Wu, C. Peng, Y. Wu, W. Zhong, J. Kang, and S. Xie, "Joint latency and charge cost minimization for reliable task offloading in dispersed computing: A multi-objective optimization approach," *IEEE Transactions on Mobile Computing*, 2026, early access.

[12] C. Peng, J. Zhang, Z. Cai, Y. Xu, Q. Huang, and H. Huang, "Strategy selection in dynamic constrained multi-objective optimization via

TABLE II

ABLATION COMPARISON OF MFT-ACO VARIANTS IN TERMS OF BEST-SO-FAR OBJECTIVE VALUES AT 600 FUNCTION EVALUATIONS ON 30 EOPMV BENCHMARK PROBLEMS (MEAN $\pm$ STD OVER 30 RUNS). BEST MEANS ARE HIGHLIGHTED IN **BOLD**.

Problem	MFT-ACO-w/o-LSBT	MFT-ACO-w/o-RBF	MFT-ACO-w/o-LS	MFT-ACO
F1	2.9037e+1 (5.22e+1) =	1.1592e+1 (3.23e+1) −	1.1171e+2 (5.53e+1) −	**5.5904e+0 (2.12e+1)**
F2	4.7408e+1 (1.66e+1) −	**2.4126e+1 (1.02e+1)** =	5.0340e+1 (1.04e+1) −	2.5356e+1 (9.48e+0)
F3	1.3601e+0 (1.48e+0) −	7.3218e+0 (4.78e+0) −	8.1754e−1 (6.34e−1) −	**2.0025e−4 (1.50e−4)**
F4	7.4170e−1 (1.58e+0) −	1.3413e+0 (4.13e+0) −	2.1401e+0 (1.26e+0) −	**1.0408e−4 (1.92e−4)**
F5	7.6406e−1 (3.10e−1) −	9.8451e−1 (2.34e+0) −	2.2446e+0 (5.68e−1) −	**5.1426e−1 (3.04e−1)**
F6	9.7222e+1 (2.60e+2) =	1.0144e+2 (3.07e+2) −	1.2831e+2 (6.59e+1) −	**2.0888e+1 (3.38e+1)**
F7	4.0317e+1 (1.47e+1) −	**2.2935e+1 (1.56e+1)** +	5.3812e+1 (8.19e+0) −	3.0771e+1 (1.20e+1)
F8	2.6962e+0 (2.37e+0) −	4.6626e+0 (4.41e+0) −	1.0620e+0 (7.38e−1) −	**2.0059e−1 (6.17e−1)**
F9	3.3650e+0 (4.14e+0) −	8.8308e−1 (2.58e+0) −	2.6814e+0 (1.53e+0) −	**9.2969e−2 (1.76e−1)**
F10	1.2815e+0 (1.74e+0) =	2.0075e+0 (3.22e+0) −	2.3945e+0 (7.60e−1) −	**7.9820e−1 (3.91e−1)**
F11	2.8842e+3 (3.76e+3) −	4.0834e+1 (1.39e+2) =	9.6973e+1 (1.90e+2) −	**2.2128e+1 (2.87e+1)**
F12	5.0318e+1 (1.22e+1) −	**2.9083e+1 (2.01e+1)** =	4.9069e+1 (1.39e+1) −	3.9683e+1 (1.58e+1)
F13	4.4606e+0 (1.48e+0) −	9.7986e−1 (2.09e+0) =	7.7643e−1 (2.21e+0) −	**1.5739e−3 (7.03e−3)**
F14	1.5071e+1 (3.11e+1) −	**2.6194e−1 (2.38e−1)** =	1.0477e+0 (1.68e+0) −	3.4900e−1 (2.95e−1)
F15	6.6776e+1 (6.19e+1) −	**6.8659e−1 (9.78e−1)** +	1.4568e+0 (1.50e+0) =	1.8773e+0 (2.80e+0)
F16	4.0962e+3 (5.83e+3) −	**1.9945e+2 (2.87e+2)** =	4.4077e+2 (6.55e+2) =	2.9424e+2 (3.77e+2)
F17	6.3555e+1 (1.07e+1) −	5.3945e+1 (1.16e+1) −	5.0158e+1 (1.26e+1) =	**4.6566e+1 (1.05e+1)**
F18	3.8519e+0 (1.41e+0) −	2.3795e+0 (2.93e+0) −	1.6092e+0 (2.31e+0) −	**4.2332e−1 (1.32e+0)**
F19	3.7940e+1 (5.80e+1) −	3.4801e+0 (6.67e+0) =	1.7359e+0 (1.26e+0) =	**1.6299e+0 (1.01e+0)**
F20	5.3405e+1 (5.51e+1) −	**2.1941e+0 (1.36e+0)** =	3.6559e+0 (1.57e+0) =	3.2344e+0 (1.86e+0)
F21	2.9272e+2 (4.99e+2) −	5.6755e+1 (1.63e+2) −	5.5118e+1 (2.78e+1) −	**3.3086e+1 (1.89e+1)**
F22	6.3541e+1 (1.35e+1) −	**3.9120e+1 (2.03e+1)** =	4.8681e+1 (1.00e+1) =	5.0427e+1 (8.74e+0)
F23	5.7887e+0 (2.69e+0) −	3.9614e+0 (4.45e+0) =	4.8013e−1 (7.98e−1) −	**2.3573e−5 (4.52e−5)**
F24	1.4455e+1 (2.10e+1) −	**1.7977e−2 (5.68e−2)** +	1.0773e+0 (6.98e−1) −	5.4407e−1 (5.60e−1)
F25	2.3680e+0 (1.72e+0) −	1.1901e+0 (2.12e+0) −	1.7501e+0 (6.57e−1) −	**6.5720e−1 (4.51e−1)**
F26	6.9837e+2 (9.15e+2) −	1.4997e+2 (1.91e+2) =	2.1480e+2 (1.82e+2) =	**1.3699e+2 (6.77e+1)**
F27	6.2939e+1 (1.38e+1) −	**4.1353e+1 (1.74e+1)** =	5.6633e+1 (7.44e+0) −	4.8160e+1 (1.18e+1)
F28	8.1850e+0 (1.83e+0) −	1.9707e+0 (2.04e+0) −	1.4287e+0 (1.11e+0) −	**2.1516e−1 (5.28e−1)**
F29	7.2285e+0 (5.03e+0) −	2.6818e+0 (3.92e+0) −	3.2087e+0 (1.27e+0) =	**2.6647e+0 (1.25e+0)**
F30	7.5499e+0 (5.43e+0) −	**1.5974e+0 (1.28e+0)** +	2.9301e+0 (2.18e+0) =	2.7721e+0 (1.51e+0)
+/ − / =	0/27/3	4/14/12	0/21/9	/

state-augmented deep reinforcement learning," *IEEE Transactions on Evolutionary Computation*, 2026, early access.

[13] 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, Aug. 2017.

[14] F. Li, X. Cai, L. Gao, and W. Shen, "A surrogate-assisted multiswarm optimization algorithm for high-dimensional computationally expensive problems," *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1390–1402, Mar. 2021.

[15] J. Liu, Y. Wang, G. Sun, and T. Pang, "Solving highly expensive optimization problems via evolutionary expected improvement," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 8, pp. 4843–4855, Aug. 2023.

[16] 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, Oct. 2020.

[17] C. D. Sutton, "Classification and regression trees, bagging, and boosting," *Handbook of statistics*, vol. 24, pp. 303–329, 2005.

[18] T. Liao, K. Socha, M. A. M. de Oca, T. Stützle, and M. Dorigo, "Ant colony optimization for mixed-variable optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 503–518, Aug. 2014.

[19] Y. Feng, L. Feng, X. Xue, S. Kwong, and K. C. Tan, "A review on evolutionary multiform transfer optimization," in *Proc. IEEE Congress on Evolutionary Computation*, 2024, pp. 1–8.

[20] Y. Feng, L. Feng, S. Kwong, and K. C. Tan, "A multiform evolutionary search paradigm for bilevel multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 28, no. 6, pp. 1719–1732, Dec. 2024.

[21] X. Xue, C. Yang, L. Feng, K. Zhang, L. Song, and K. C. Tan, "Solution transfer in evolutionary optimization: An empirical study on sequential transfer," *IEEE Transactions on Evolutionary Computation*, vol. 28, no. 6, pp. 1776–1793, Dec. 2024.

[22] X. Xue, Y. Hu, L. Feng, K. Zhang, L. Song, and K. C. Tan, "Surrogate-assisted search with competitive knowledge transfer for expensive optimization," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 6, pp. 2416–2430, Dec. 2025.

[23] S.-H. Wu, Y. Huang, X. Wu, L. Feng, Z.-H. Zhan, and K. C. Tan, "Learning to transfer for evolutionary multitasking," *IEEE Transactions on Cybernetics*, vol. 55, no. 7, pp. 3342–3355, Jul. 2025.

[24] S.-H. Wu, Z.-H. Zhan, K. C. Tan, and J. Zhang, "Transferable adaptive differential evolution for many-task optimization," *IEEE Transactions on Cybernetics*, vol. 53, no. 11, pp. 7295–7308, Nov. 2023.

[25] Y. Feng, L. Feng, S. Kwong, and K. C. Tan, "A multivariation multifactorial evolutionary algorithm for large-scale multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 2, pp. 248–262, Apr. 2022.

[26] C. Yang, J. Ding, Y. Jin, and T. Chai, "Offline data-driven multiobjective optimization: Knowledge transfer between surrogates and generation of final solutions," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 409–423, Jun. 2020.

[27] K. Qiao, K. Yu, B. Qu, J. Liang, H. Song, and C. Yue, "An evolutionary multitasking optimization framework for constrained multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 2, pp. 263–277, Apr. 2022.

[28] H. Gu, H. Wang, and Y. Jin, "Surrogate-assisted differential evolution with adaptive multisubspace search for large-scale expensive optimization," *IEEE Transactions on Evolutionary Computation*, vol. 27, no. 6, pp. 1765–1779, Dec. 2023.

[29] K. Nishihara and M. Nakata, "Surrogate-assisted differential evolution with adaptation of training data selection criterion," in *Proc. IEEE Symposium Series on Computational Intelligence*, 2022, pp. 1675–1682.

[30] G. Chen, K. Zhang, X. Xue, L. Zhang, J. Yao, H. Sun, L. Fan, and Y. Yang, "Surrogate-assisted evolutionary algorithm with dimensionality reduction method for water flooding production optimization," *Journal of Petroleum Science and Engineering*, vol. 185, p. 106633, 2020.