Multioutput Surrogate Assisted Evolutionary Algorithm for Expensive Multi-Modal Optimization Problems

1st Renzhi Chen
Defense Innovation Institute
Beijing, China
chenrenzhi1989@gmail.com

2nd Ke Li
Department of Computer Science
University of Exeter
Exeter, EX4 4QF, UK
k.li@exeter.ac.uk

***Abstract*—Real-world optimization problems are often computationally expensive and feature multi-modal objective functions. Surrogate-assisted evolutionary optimization has proven to be an effective approach for addressing expensive black-box optimization challenges, but the technique has not been adequately studied in multi-modal situations. In this paper, we propose a simple but effective multi-output surrogate-based approach for empowering surrogate-assisted evolutionary optimization to address expensive multi-modal optimization problems. Specifically, our proposed approach employs a multi-output Gaussian process to capture correlations between data collected from different local areas. Experiments on synthetic benchmark test problems demonstrate the effectiveness of our proposed algorithm against six state-of-the-art peer algorithms.**

***Index Terms*—Multi-Modal Optimization, Surrogate-Assisted Evolutionary Algorithm, Multi-Output Gaussian Process**

I. Introduction

Real-world optimization problems usually involve computationally or economically expensive objective function(s). For instance, evaluating the performance of a hardware accelerator for a general-purpose matrix multiplier (GEMM) through a single function evaluation (FE) usually take hours [1]. Another intricate feature is the presence of numerous local optima that can impede the convergence towards the global optimum, as known as multi-modal optimization problems (MMOPs). Such problems are prevalent across various domains, including electromagnetic system design [2], task scheduling [3], and neural network ensembles [4]. In these scenarios, accurately identifying the global optimum is of paramount importance for decision-making, ensuring optimal performance, and facilitating efficient resource allocation. The unique properties of MMOPs necessitate the development of specialized optimization algorithms that are capable of effectively navigating the search space, balancing exploration and exploitation, and avoiding entrapment in local optima. This becomes increasingly important and challenging when FEs are expensive. Such optimization problems are commonly referred to as expensive MMOPs in literature [5]–[7].

There have been several studies proposing surrogate-assisted evolutionary algorithms (SAEAs) that can be directly applied to solve expensive MMOPs. These algorithms utilize a global surrogate model, such as polynomial regression (PR) [8], radial basis function networks (RBF) [9] or Gaussian processes (GP) [10], to fit samples collected throughout the entire search space. In order to improve the performance of these SAEAs, some studies have also focused on the preprocessing of data samples. For example, Liu et al. [10] employed an embedding method prior to building the GP model in their SAEA. Wang et al. [11] utilized a domain selection mechanism to choose proper data sets for fitting.

Although these algorithms have shown promising results in solving optimization problems with expensive FEs, they are susceptible to getting stuck in local optima due to the absence of effective multimodal processing mechanisms. To address this issue, a common solution is to replace the global surrogate model with an ensemble of surrogate models. Several researchers have focused on constructing an ensemble of surrogate models designed for both the entire search space and local regions. Lim et al. [12] built surrogates around local optimal solutions, whereas Yahyaie and Filizadeh [13] used an adaptive mesh to identify regions that may include local optimal solutions. Ji et al. [6] used dual-layer surrogate models consists of an upper global surrogate model and a set of lower local surrogate models. Gao et al. [7] divides optimization into two stages: first detects promising subregions, and then performs local searches in each subregion based on a local surrogate model. Some others built multiple different surrogates for the entire search space. Dong et al. [14] proposed a multiple-surrogate-based optimization algorithm by integrating Kriging, RBF, and PR surrogate models into differential evolution. Similarly, Ji et al. [15] constructed three surrogate models but optimize based on a multi-task EAs. These existing works mainly have three limitations as follows.

- Most of the proposed methods rely on distance-based techniques, such as k-means clustering [7] or hill-valley methods [16], to divide the search space when building local surrogate models. However, due to the characteristics of the optimization process, the evaluated samples are frequently unevenly distributed across the search space, which can result in certain local surrogate models lacking

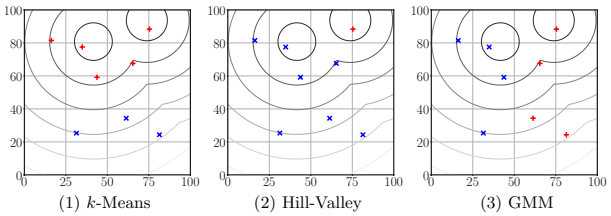

Fig. 1. A comparison of clustering methods applied to a sample dataset evaluated over problems with two optima.

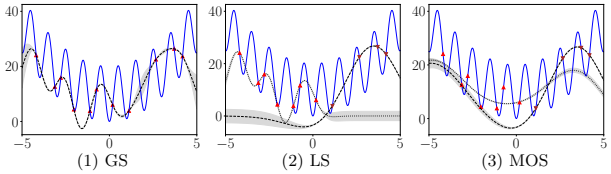

Fig. 2. An illustrative example of the underfitting problem of using (1) the global surrogate (GS), (2) the local surrogates (LS) and (3) the multi-output surrogate (MOS) for Rastrigin function ($n = 1$).

the necessary amount of data to fit accurately, potentially leading to poor algorithm performance. We show a simple example clustering samples evaluated over a function with two optima. Both distance-based method perform poorly. The k-means method, as shown in Figure 1(1), wrongly clustered the samples into two sets, resulting in poor performance. The hill-valley method, as shown in Figure 1(2), was able to correctly distinguish samples for one hill from the other, but this led to an unbalanced split where one cluster had most of the samples and the other had only one sample.

- The importance of surrogate modelling itself is often overlooked. Although global surrogate models offer a broader perspective on the optimization problem, they may fail to capture intricate features of the landscape. In contrast, local surrogate models are more likely to produce accurate fitness estimates but can be prone to producing misleading results when trained on limited data, which has not been sufficiently addressed in literature. To further illustrate this issue, we consider an illustrative example of a Rastrigin function with $n = 1$. The figure demonstrates that the global surrogate (GS) in Figure 2(1) and the local surrogates (LS) in Figure 2(2). The GS fails due to limited samples over a complex landscape, while the LS fails because there is insufficient local sample data to adequately fit a proper Gaussian process (GP).
- Since each surrogate in the ensemble will have its own suggested optimal solutions, existing approaches often evaluate all these solutions. However, in the case of expensive FEs, this presents a significant waste of resources, particularly as the number of surrogate models increases.

To address the limitations of existing algorithms designed for expensive MMOPs, this paper proposes a novel approach called the Multi-Output Surrogate Assisted Evolutionary Algorithm (`MSAEA`) which consists of three distinct components:

- `MSAEA` adopts a distribution-based clustering method, Gaussian mixture model (GMM), to divide the population into sub-populations that represent different local areas, instead of relying on distance-based clustering.
- In contrast to using multiple local surrogate models, `MSAEA` uses a global multi-output surrogate model with each output representing a local area that captures the corresponding local landscape. Additionally, the model can share information between different local areas via covariance.
- `MSAEA` utilizes a hypervolume contribution-based selection mechanism to propose the candidate for evaluation on the expensive objective function over multi-output surrogate model.

The remainder of this paper is organized as follows: In Section II, we provide a detailed explanation of our proposed algorithm, `MSAEA`. We present the experimental setup of the empirical studies in Section IV. Then, in Section IV, we compare and analyze the results obtained. Finally, in Section V, we summarize our findings and conclusions.

II. Proposed Algorithm

The flowchart of `MSAEA` is given in Fig. 3. It starts with an `initialization` step based on Latin hypercube sampling without loss of generality where the initial population $\mathcal{P}_0 = \{(\mathbf{x}^i, f(\mathbf{x}^i))\}_{i=1}^N$ is generated. For t generation, we first evaluate the objective function values of samples which are not evaluated in the population $\mathcal{P}_t$. As for the other three following steps, we will delineate their implementations in the following subsection.

A. GMM Clustering

The `GMM clustering` step aims to split the population $\mathcal{P}_t = \{(\mathbf{x}^i, f(\mathbf{x}^i))\}_{i=1}^N$ into C sub-populations $\{P_t^1, \cdots, P_t^C\}$ which representing different local areas, where $P_t^c = \{(\langle \mathbf{x}^i, c\rangle, f(\mathbf{x}^i))\}_{i=1}^{N^c}$. GMM fits the distribution of the population using a weighted sum of Gaussian distributions, with each Gaussian distribution representing a subpopulation. The parameters of the Gaussian distributions Θ, which includes the means μ_c, covariances Σ_c, and mixture weights ω_c for $c \in 1, \cdots, C$, are fit from the coordinates in the decision space using the Expectation-Maximization (EM) algorithm. The pseudo-code for fitting the GMM is given in Algorithm 1. Once GMM fitting is complete, each subpopulation selects individuals based on their density, i.e., the probability of they belong to the local area represented by this subpopulation.

A simple example demonstrates the advantages of using GMM. Figure 2(3) shows how GMM split the samples into two subpopulations with equal numbers of samples.

B. Multi-output Surrogate modeling

The goal of `multi-output surrogate modelling` step is building a multi-output Gaussian process (MOGP) regression model, where each output captures the local landscape and shares information via covariance between outputs.

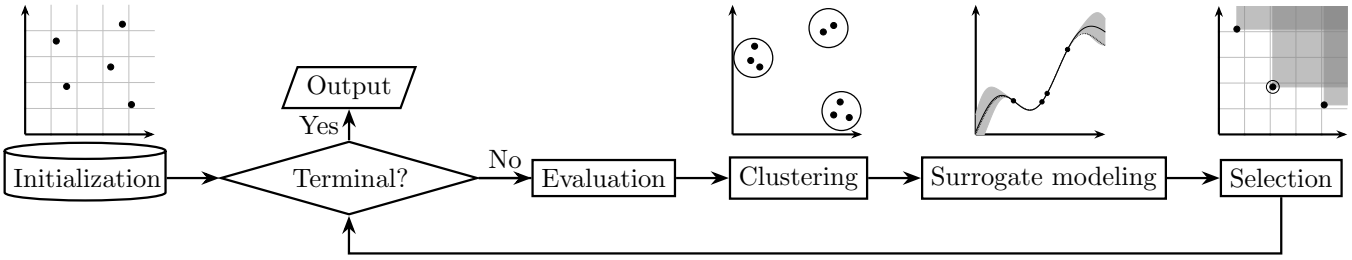

Fig. 3. Flowchart of MSAEA .

Algorithm 1: GMM fitting

Input: The dataset X, Number of clusters C, threshold ϵ, parameters for previous fited GMM $\hat{\Theta}$
Output: Posterior Density Matrix $\mathcal{D}$
Initialization: If $\hat{\Theta}$ is empty then generate Θ randomly, else $\Theta \leftarrow \hat{\Theta}$;
repeat
Expectation step: Calculate the posterior density Matrix $\mathcal{D}$ according to:

$$d_{rc} \leftarrow \frac{\omega_c \mathcal{N}(\mathbf{x}^r|\mu_c, \Sigma_c)}{\sum_{j=1}^{C} \omega_j \mathcal{N}(\mathbf{x}^r|\mu_j, \Sigma_j)}$$

for $r \in \{1, \cdots, N\}$, $c \in \{1, \cdots, C\}$;
Maximization step: Update Θ:

$$\begin{aligned} N_c &\leftarrow \textstyle\sum_{i=1}^{N} d_{ic} \\ \mu_c &\leftarrow \tfrac{1}{N_c} \textstyle\sum_{i=1}^{N} d_{ic}\mathbf{x}^i \\ \Sigma_c &\leftarrow \tfrac{1}{N_c} \textstyle\sum_{i=1}^{N} d_{ic}(\mathbf{x}^i - \mu_c)(\mathbf{x}^i - \mu_c)^T \\ \omega_c &\leftarrow N_c/N \end{aligned}$$

for $c \in \{1, \cdots, C\}$;
Check for convergence:

$$\begin{aligned} L_{old} &\leftarrow L \\ L &\leftarrow \log p(X|\Theta) \end{aligned}$$

until $|L - L_{old}| < \epsilon$;

Given the set of subpopulation $\{P_t^1, \cdots, P_t^C\}$, for each testing sample $\langle \mathbf{x}^*, c\rangle$, the mean and variance are predicted as:

$$\begin{aligned} \mu\big(\langle \mathbf{x}^*, c\rangle\big) &= {\mathbf{k}^*}^\top K^{-1}\mathbf{f} \\ \sigma\big(\langle \mathbf{x}^*, c\rangle\big) &= k(\mathbf{x}^*, \mathbf{x}^*) - {\mathbf{k}^*}^\top K^{-1}\mathbf{k}^*, \end{aligned} \tag{1}$$

where the covariance function here we adapt the Linear Model of Coregionalization (LMC) kernel, which is defined as:

$$k\big(\langle \mathbf{x}, c\rangle, \langle \mathbf{x}', c'\rangle\big) = \sum_{q=1}^{Q} [A_q]_{c,c'} k_q(\mathbf{x}, \mathbf{x}') \tag{2}$$

where $\langle \mathbf{x}, c\rangle$ represents a sample $\mathbf{x}$ for the c-th output, and $\langle \mathbf{x}', c'\rangle$ represents another sample $\mathbf{x}'$ from the c'-th output. Here, Q is a hyperparameter that indicates the number of shared kernels, and in our case, we set $Q = C$. The A_q matrix is a $C \times C$ matrix, where $[A_q]_{c,c'}$ represents the q-th scaling factor between the c-th output and the c'-th output. The q-th shared kernel function, $k_q(\mathbf{x}, \mathbf{x}')$, measures the similarity between inputs $\mathbf{x}$ and $\mathbf{x}'$. Here we adapt RBF kernel for $k_q(\mathbf{x}, \mathbf{x}')$.

To further illustrate this issue, we consider an illustrative example of a Rastrigin function with $n = 1$ in Figure 2(3), the multi-output surrogate (MOS) adapted in our algorithm successfully captures the true landscape of the test function in both of its outputs. Moreover, these two outputs have varying levels of uncertainty due to the differences in the data represented.

C. Selection

Based on the MOGP build in step `multi-output surrogate modelling`, we adapt the Lower Confidence Bound (LCB) acquisition function to propose a vector of candidates for each output. The LCB acquisition function for MOGP is defined as:

$$\boldsymbol{\alpha}_{\text{LCB}}(\mathbf{x}^*) = \boldsymbol{\mu}(\mathbf{x}^*) - \kappa\boldsymbol{\sigma}(\mathbf{x}^*) \tag{3}$$

where $\boldsymbol{\mu}(\mathbf{x}^*) = \big(\mu(\langle \mathbf{x}^*, 1\rangle), \cdots, \mu(\langle \mathbf{x}^*, C\rangle)\big)^\top$ and $\boldsymbol{\sigma}(\mathbf{x}^*) = \big(\sigma(\langle \mathbf{x}^*, 1\rangle), \cdots, \sigma(\langle \mathbf{x}^*, C\rangle)\big)^\top$ are the vector of predicted mean and variance at test sample $\mathbf{x}^*$ respectively, and κ is a trade-off parameter that balances the exploitation and exploration in the optimization process. The proposed candidate is select base on the hypervolume of $\boldsymbol{\alpha}_{\text{LCB}}(\mathbf{x})$:

$$\mathbf{x}^\star = \operatorname*{argmin}_{\mathbf{x}} \text{HV}(\boldsymbol{\alpha}_{\text{LCB}}(\mathbf{x})) \tag{4}$$

where the hypervolume is defined as

$$\text{HV}(\boldsymbol{\alpha}) = \int_{-\infty}^{\infty} \prod_{c=1}^{C} \Big(\alpha_i - \alpha_i^*\Big) \mathrm{d}\mathbf{x} \tag{5}$$

in which $\boldsymbol{\alpha}_{\text{LCB}}(\mathbf{x}^*) = (\alpha_1, \cdots, \alpha_C)^\top$ and $\boldsymbol{\alpha}^* = (\alpha_1^*, \cdots, \alpha_C^*)^\top$ is a reference point, typically the ideal point. We use differential evolution (DE) [17] here to search for the candidate.

III. Experimental Setup

The performance of MSAEA is evaluated and compared with other peer algorithms on synthetic benchmark test problems. The following paragraphs will briefly introduce the benchmark problems, peer algorithms and performance measures.

A. Synthetic Benchmark Problem

We choose three classic multimodal optimization problems as the synthetic benchmark problem in our experiments.

- **Rastrigin Function** has several local minima. It is highly multimodal, but locations of the minima are regularly distributed.
- **Ackley Function** is widely used for testing optimization algorithms which are characterized by a nearly flat outer region, and a large hole at the centre.
- **Griewank Function** has many widespread local minima, which are regularly distributed.

The mathematical definition and search bound are given in Section I-A of the supplementary file.

B. Peer Algorithms

In our empirical study, the performance of our proposed MSAEA is compared with the other six peer algorithms.

- Bayesian Optimization (BO): an approach to global optimization that utilizes a probabilistic model to estimate the location of the next point to evaluate on a black-box objective function. The model is updated using previous evaluations, and acquisition functions are used to guide the search towards the global optimum.
- BO with clustering (cBO) [18]: a variation BO which divides the dataset into several subsets using clustering techniques.
- Decomposition differential evolution based on RBF surrogate (D/REM) [7]: a two-phase algorithm first decomposes EMMOPs into subproblems to detect promising subregions, and then performs a local search in those subregions.
- Multisurrogate-Assisted Multitasking Particle Swarm Optimization (MaMPSO) [15]: the algorithm employs Gaussian Process (GP), Radial Basis Function (RBF), and Polynomial Regression (PR) models to model the objective function and utilizes a multi-task PSO to search for candidate solutions.
- Neighborhood Evolutionary Sampling Framework with Dynamic Repulsion (DR-NESO) [19]: it is an optimization algorithm that uses neighbourhood evolutionary sampling strategies to generate candidate solutions for real fitness evaluation. It also incorporates a taboo mechanism to avoid over-exploitation.

The parameter settings for our proposed MSAEA and the peers listed above are given in Section X in supplementary. Also, the source code for our proposed MSAEA and the peers listed above is in https://github.com/RenzhiChen/MSAEA.

C. Performance Measures

In this paper, we consider the following two performance metrics.

- Average error overtime: it measures the average deviation from the true global optimum overall FEs:

$$\bar{\epsilon}_{\mathrm{f}} = \frac{1}{N_{\mathrm{FE}}} \sum_{i=1}^{N_{\mathrm{FE}}} \Big[f(\mathbf{x}^*) - f_i(\bar{\mathbf{x}}^*) \Big], \tag{6}$$

  where N_{FE} is the total number of FEs, $f(\mathbf{x}^*)$ is the global optimum and $f_i(\bar{\mathbf{x}}^*)$ is the best solution found at the i-th FE.
- Overall error: it is the final accuracy achieved at the end of the optimization process:

$$\bar{\epsilon}_{\mathrm{t}} = f(\mathbf{x}^*) - f(\hat{\mathbf{x}}^*), \tag{7}$$

  where $f(\mathbf{x}^*)$ is the true global optimum and $f(\hat{\mathbf{x}}^*)$ is the best solution found at the end of optimization.

To have a statistical interpretation of the significance of comparison results, we use the Wilcoxon signed-rank test [20], Scott-Knott test [21] and A_{12} [22] effect size in our empirical study. The detail of these statistical tests is given in Section I-B in the supplementary file.

D. Parameter Settings

The parameter settings used in our experiments are summarized as follows.

- Test problem settings: We consider different number of variables as $n \in \{3, 5, 8, 10\}$.
- Computational budget: In our experiments, we consider $N_{\mathrm{FE}} = 11n - 1$ with $11n - 1$ for initial sampling.
- Parameters of peer algorithms: The parameters of MSAEA and peer algorithms are given in Section I-C of the supplementary file.

IV. Results and Analysis

This section evaluates and analyzes the effectiveness of MSAEA by answering the following research questions (RQs) on synthetic problems.

- **RQ1:** How is the performance of MSAEA compared to the peer algorithms for different dimensions?
- **RQ2:** How is the performance of MSAEA compared to the peer algorithms for the different number of optimal?
- **RQ3:** In case there are positive results for RQ1 and RQ2, why does MSAEA work?

A. Comparison with the Peer Algorithms

In this subsection, we investigate the performance comparison of our proposed MSAEA against the other peer algorithms.

1) Methods: To address RQ1, the results are presented from the following aspects.

- First, the statistical comparison results on $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ are given in Table I of the supplementary file. From the comparison results on $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$, we can see that our proposed MSAEA is the best algorithm in most of the comparisons. Note that all the better results are of statistical significance according to the Wilcoxon signed-rank test.
- In view of the overwhelmingly superior performance observed above, it is not surprising to see the stacked bar charts of A_{12} shown in Fig. 4 demonstrate that most the better comparison results achieved by MSAEA are always classified to be large.
- Likewise, based on the previous two comparison results, it is not difficult to infer the outstanding rank achieved

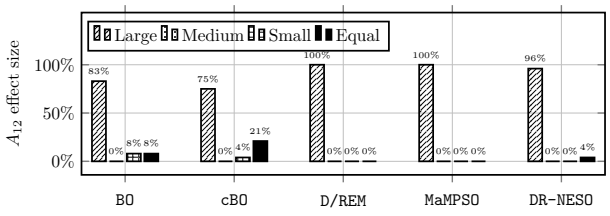

Fig. 4. Percentage of the large, medium, small, and equal A_{12} effect size of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ when comparing MSAEA with the other five peer algorithms on Ackley, Rastrigin and Griewank functions.

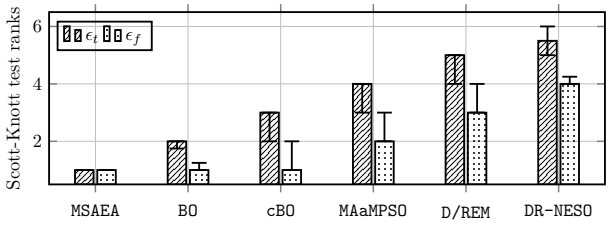

Fig. 5. Bar charts with error bars of Scott-Knott test ranks of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ achieved by each of the six algorithms on Ackley, Rastrigin and Griewank functions (the smaller rank is, the better performance achieves).

by MSAEA according to the Scott-Knott test as shown in Figs. 5.

- From the selected trajectories of $\mathcal{L}(\mathbf{x})$ shown in Figs. 6 while full results are provided in Figs. 1 of the supplemental document, we can see that the performance MSAEA is better than the other five peer algorithms. This can be attributed to the effectiveness of leveraging the sharing landscape 'knowledge' among the local areas. Although the other peer algorithms, except BO, claimed to have a strategy to handle multimodality, they are not as effective as MSAEA. The results in some cases are worse than BO, which builds a single global surrogate. This might be because these algorithms although claimed to be developed for expensive scenarios, still require a number of solutions to support a proper fit for local surrogates. Since only a limited amount of data are collected, they are not adequate to train a good model locally.

B. Comparison with the Peer Algorithms for Different Numbers of Optimal

In this subsection, we investigate the effectiveness of MSAEA with a different number of optimal, which is mentioned in RQ2. To address RQ2, as the number of local optimums for the Ackley function can be adjusted by given different c, we design the following three tests with $c = \pi, 2\pi, 4\pi$ respectively. Let us interpret the results from the experiments for RQ2.

- The statistical comparison results on $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ are given in Table II of the supplementary file. From the comparison results we can see that our proposed MSAEA is the best algorithm in most of the comparisons. Note that all the better results are of statistical significance according to the Wilcoxon signed-rank test.
- In view of the overwhelmingly superior performance observed above, it is not surprised to see the stacked bar charts of A_{12} shown in Fig. 7 demonstrate that most the better comparison results achieved by MSAEA are always classified to be large.
- Likewise, based on the previous two comparison results, it is not difficult to infer the outstanding rank achieved by MSAEA according to the Scott-Knott test as shown in Figs. 8.
- From the trajectories of $\mathcal{L}(\mathbf{x})$ shown in Figs. 2 of the supplemental document, we can see that the performance MSAEA is better than the other five peer algorithms when the number of local optima changes.

C. Why Does MSAEA Work

In this subsection, we perform an ablation study to investigate the effectiveness of GMM and the multi-output surrogate of MSAEA . To address RQ3, we develop three variants with different surrogates.

- MSAEA-v1: it replaces MOGP with multiple local GP surrogates.
- MSAEA-v2: it replaces MOGP with a single global GP surrogate.
- MSAEA-v3: it replaces GMM with k-means clustering.

Based on the comparison results of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ in Table III of the supplementary file, there appears to be a significant performance degradation of MSAEA when the MOGP is replaced with other surrogate(s). In view of the overwhelmingly superior performance in the table, it is not surprised to see the stacked bar charts of A_{12} shown in Fig. 9 demonstrate that most the better comparison results achieved by MSAEA are always classified to be large. This is counter-intuitive at first glance, as the single global surrogate model typically performs worse than a set of local surrogate models. However, in expensive search scenarios, splitting limited solutions into subgroups leads to a lack of data for surrogates to accurately fit.

V. Conclusions

This paper develops a simple but effective multi-output surrogate-based method that empowers surrogate-assisted evolutionary optimization to tackle expensive multi-modal black-box optimization problems. Existing MOGPs have limited scalability when dealing with numerous tasks. However, real-world systems are typically constructed to address problems with a large number of local optima. One of our following goals is to create a multi-output surrogate capable of handling multiple outputs.

References

[1] L. Nardi, D. Koeplinger, and K. Olukotun, "Practical design space exploration," in *27th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, MASCOTS 2019, Rennes, France, October 21-25, 2019*. IEEE Computer Society, 2019, pp. 347–358.

[2] C.-H. Yoo, D.-K. Lim, and H.-K. Jung, "A novel multimodal optimization algorithm for the design of electromagnetic machines," *IEEE Transactions on Magnetics*, vol. 52, no. 3, pp. 1–4, 2015.

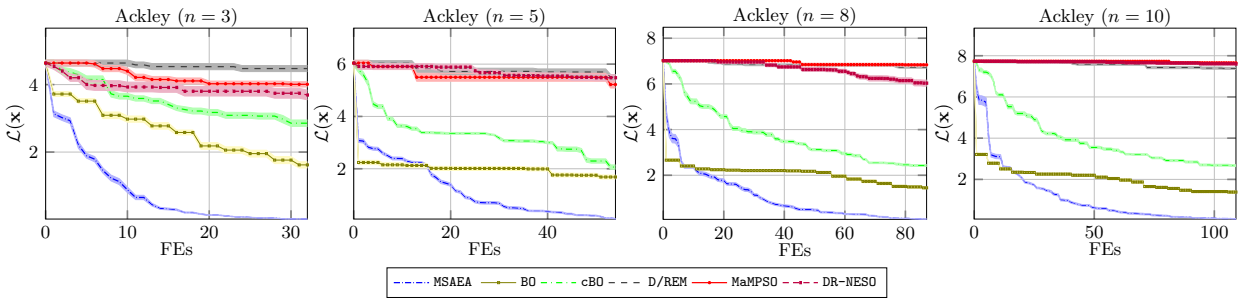

Fig. 6. Trajectories of $\mathcal{L}(\mathbf{x})$ with confidence bounds achieved by six algorithms on Ackley functions.

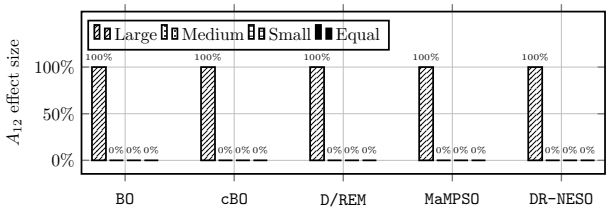

Fig. 7. Percentage of the large, medium, small, and equal A_{12} effect size of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ when comparing MSAEA with the other five peer algorithms on Ackley function given different c.

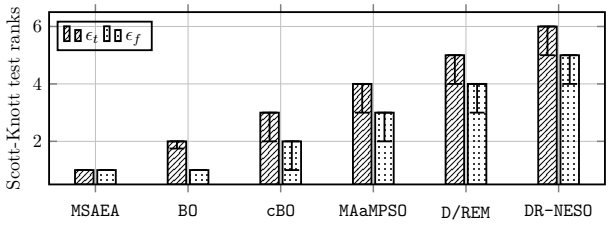

Fig. 8. Bar charts with error bars of Scott-Knott test ranks of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ achieved by each of the ten algorithms on Ackley function given different c (the smaller rank is, the better performance achieves).

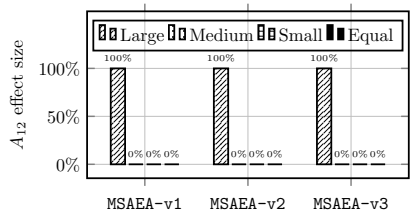

Fig. 9. Percentage of the large, medium, small, and equal A_{12} effect size of $\bar{\epsilon}_{\mathrm{f}}$ and $\bar{\epsilon}_{\mathrm{t}}$ when comparing MSAEA with the other three variants.

[3] E. Pérez, M. Posada, and A. Lorenzana, "Taking advantage of solving the resource constrained multi-project scheduling problems using multi-modal genetic algorithms," *Soft Comput.*, vol. 20, no. 5, pp. 1879–1896, 2016.

[4] Y. Gong, J. Zhang, and Y. Zhou, "Learning multimodal parameters: A bare-bones niching differential evolution approach," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 7, pp. 2944–2959, 2018.

[5] H. Dong, B. Song, P. Wang, and Z. Dong, "Surrogate-based optimization with clustering-based space exploration for expensive multimodal problems," *Structural and Multidisciplinary Optimization*, vol. 57, pp. 1553–1577, 2018.

[6] X. Ji, Y. Zhang, D. Gong, and X. Sun, "Dual-surrogate-assisted cooperative particle swarm optimization for expensive multimodal problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 794–808, 2021.

[7] W. Gao, Z. Wei, M. Gong, and G. G. Yen, "Solving expensive multi-modal optimization problem by a decomposition differential evolution algorithm," *IEEE Trans. Cybern.*, vol. 53, no. 4, pp. 2236–2246, 2023.

[8] W. Roux, N. Stander, and R. THaftka, "Response surface approximations for structural optimization," *International journal for numerical methods in engineering*, vol. 42, no. 3, pp. 517–534, 1998.

[9] R. G. Regis, "Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 326–347, 2014.

[10] B. Liu, Q. Zhang, and G. G. E. Gielen, "A gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 180–192, 2014.

[11] X. Wang, Y. Jin, S. Schmitt, M. Olhofer, and R. Allmendinger, "Transfer learning based surrogate assisted evolutionary bi-objective optimization for objectives with different evaluation times," *Knowl. Based Syst.*, vol. 227, p. 107190, 2021.

[12] D. Lim, Y. Jin, Y. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 329–355, 2010.

[13] F. Yahyaie and S. Filizadeh, "A surrogate-model based multi-modal optimization algorithm," *Engineering Optimization*, vol. 43, no. 7, pp. 779–799, 2011.

[14] H. Dong, C. Li, B. Song, and P. Wang, "Multi-surrogate-based differential evolution with multi-start exploration (MDEME) for computationally expensive optimization," *Adv. Eng. Softw.*, vol. 123, pp. 62–76, 2018. [Online]. Available: https://doi.org/10.1016/j.advengsoft.2018.06.001

[15] X. Ji, Y. Zhang, D. Gong, X. Sun, and Y. Guo, "Multisurrogate-assisted multitasking particle swarm optimization for expensive multimodal problems," *IEEE Trans. Cybern.*, vol. 53, no. 4, pp. 2516–2530, 2023.

[16] Y. Li, L. Huang, W. Gao, Z. Wei, T. Huang, J. Xu, and M. Gong, "History information-based hill-valley technique for multimodal optimization problems," *Inf. Sci.*, vol. 631, pp. 15–30, 2023.

[17] R. Storn and K. V. Price, "Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[18] I. Roman, A. Mendiburu, R. Santana, and J. A. Lozano, "Bayesian optimization approaches for massively multi-modal problems," in *Learning and Intelligent Optimization - 13th International Conference, LION 13, Chania, Crete, Greece, May 27-31, 2019, Revised Selected Papers*, ser. Lecture Notes in Computer Science, N. F. Matsatsinis, Y. Marinakis, and P. M. Pardalos, Eds., vol. 11968. Springer, 2019, pp. 383–397.

[19] H. Zhen, S. Xiong, W. Gong, and L. Wang, "Neighborhood evolutionary sampling with dynamic repulsion for expensive multimodal optimization," *Inf. Sci.*, vol. 630, pp. 82–97, 2023.

[20] F. Wilcoxon, "Individual comparisons by ranking methods," 1945.

[21] N. Mittas and L. Angelis, "Ranking and clustering software cost estimation models through a multiple comparisons algorithm," *IEEE Trans. Software Eng.*, vol. 39, no. 4, pp. 537–551, 2013.

[22] A. Vargha and H. D. Delaney, "A critique and improvement of the CL common language effect size statistics of McGraw and Wong," *J. Educ. Behav. Stat.*, vol. 25, no. 2, pp. 101–132, 2000.