IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 21 January 2023, accepted 5 February 2023, date of publication 14 February 2023, date of current version 23 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3244996

RESEARCH ARTICLE

An Adaptive Parallel EI Infilling Strategy Extended by Non-Parametric PMC Sampling Scheme for Efficient Global Optimization

YU HU[1], YAOLIN GUO[2], ZHEN LIU[3], YIFAN LI[3], ZHEYU HU[1], DIWEI SHI[4], MORAN BU[2,7], AND SHIYU DU[2,5,6]

[1]Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China
[2]Engineering Laboratory of Advanced Energy Materials, Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo 315201, China
[3]College of Material Science and Chemical Engineering, Harbin Engineering University, Heilongjiang 150006, China
[4]School of Naval Architecture and Maritime, Zhejiang Ocean University, Zhejiang 316022, China
[5]School of Materials Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China
[6]School of Computer Science, China University of Petroleum (East China), Qingdao 266580, China
[7]Key Laboratory of Soybean Molecular Design Breeding, Northeast Institute of Geography and Agroecology, Chinese Academy of Sciences, Changchun 130102, China

Corresponding authors: Shiyu Du (dushiyu@nimte.ac.cn) and Yaolin Guo (guoyaolin@nimte.ac.cn)

the National Key R&D Program of China (Grant No. 2019YFB1901001), the Key R&D Projects of Zhejiang Province (No. 2022C01236, 2019C01060), the National Natural Science Foundations of China (Grant Nos. 21875271, 12105066, U20B2021, 21707147, 51372046, 51479037, 91226202, and 91426304), the National Key Research and Development Program of China (No. 2016YFB0700100), the Entrepreneuship Program of Foshan National Hi-tech Industrial Development Zone, the Major Project of the Ministry of Science and Technology of China (Grant No. 2015ZX06004-001), Ningbo Natural Science Foundations (Grant Nos. 2019A610106).

ABSTRACT This paper presents a novel adaptive parallel Expected Improvement (EI) infilling strategy for Efficient Global Optimization (EGO) by introducing a two-staged Non-parametric Population Monte Carlo Sampling (NPMS) scheme. The samples are uniformly generated from EI function in the first stage and converge to sub-domains of high EI values thresholded by a non-parametric sampling selection method in Population Monte Carlo (PMC) iterative succession. In the second stage, learning from potential information, Density-Based Spatial Clustering (DBSCAN) method is used to cluster samples and converge to candidate points. Compared to the original EI strategy, NPMS improves the minimum result by 14.6% and reduces the number of candidate points by 15.8% on our benchmark cases of EGO. Furthermore, 13 test functions involving different input space sizes, difficulties, and dimensions are conducted on six strategies including NPMS, and the results showed that NPMS achieves the highest ranking in terms of result finding and cost savings but slightly decreases optimization efficiency. Benefiting from broad sampling and dynamic clustering, especially in large input space size cases, NPMS not only guarantees high result accuracy but also reduces optimization costs by up to 34.9% compared to other parallel methods. Finally, our proposed NPMS-extended EI strategy has successfully reduced the number of candidate points, which is expected to provide a cost-practical approach to more complex problems.

INDEX TERMS Expected improvement, multi-peak characteristics, parallel infilling strategy, population Monte Carlo, sampling method.

I. INTRODUCTION

Efficient Global Optimization (EGO) is a model-based sequential optimization technique that effectively solves the classical machine intelligence problem in decision theory [1], [2], [3], [4]. EGO is based on the kriging model and Expected Improvement (EI) infilling strategy. Kriging model provides cheap evaluation (prediction $\mu(x)$ and uncertainty measure $\sigma(x)$). And EI infilling strategy provides promising candidate points (maximum EI function value),

The associate editor coordinating the review of this manuscript and approving it for publication was Kuo-Ching Ying.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

which effectively addresses the balance between optimization results and optimization costs in decision making [5], [6], [7]. Since proposed by Jones et al. [8] in 1998, EGO has been used in complex engineering. References [9], [10], [11], [12], [13], and [14] and has become the most popular approach. As a candidate point selection credential, infilling strategy determines the kriging performance and EGO efficiency. Effective infilling strategies based on different criteria are proposed, such as the probability of improvement (PI) [15], goal seeking [16], upper confidence bound (UCB) [17]. Among them, the most famous and widely used is EI infilling strategy due to a good balance between global exploration and local exploitation in EGO [18], [19].

Since EI function is a greedy improvement of heuristics, EI-based EGO often falls into unbalanced problems such as over-exploration and over-exploitation, resulting in slow optimization convergence [20], [21], [22], which requires infilling strategies to make further enhancements to balance and sampling efficiency. Schonlau et al. [23] and Sóbester et al. [24] introduced additional parameters in EI function, and proposed the Generalized Expected Improvement (GEI) and the Weighted Expected Improvement (WEI), respectively. Recently, Lv et al. [25] made up for the limitation of EI strategy by mixing multiple individual infilling strategies and proposed a Go-Inspired Hybrid (GO-HI) infilling strategy. Compared with independent infilling strategies, GO-HI saves computational costs and generates global points. However, single-point infilling strategy is not economically viable for the multiple CPU cores of modern computers, especially if the simulations or physical experiments are significant time consuming [26], [27], [28], [29].

Parallel infilling strategy focuses on inefficiency of single-point selection and generates multi-point in each optimization iteration. Although sampling accuracy is inevitably reduced, it improves computation utilization and convergence speed [30]. Based on different multi-point selection schemes, researchers developed many parallel strategies. Sóbester et al. [31] proposed a parallel strategy to select n local maxima of EI as new candidate points assuming given n processors. Compared with single-point strategy, multi-point of EI function local maxima can build an effective global kriging model faster. Ginsbourger et al. [32] derived $q-$EI based on the work by Schonlau et al. [26] and pointed out that $q-$EI strategy needs to be optimized one time to produce q candidate points in each iteration. Still, the dimension of optimization problem increases to $q \times d$ (d is the dimension of the design variables). Researchers relaxed the exact EI multi-peak calculation and replaced it with fake values or the points from high EI values sub-domains. Ginsbourger et al. [33] proposed a heuristic strategy, named Kriging Believer (KB) strategy, which uses the kriging prediction replaces the actual calculation and cycle q times in a turn to obtain q samples. The KB strategy depends on the performance of the kriging model and produces candidate points clustered around false values. To solve this problem, Ginsbourger et al. [33] used a constant to replace the actual calculation of the selected samples named the Constant Liar (CL) strategy. However, low observation points lead to high uncertainty in constant selection. Inspired by the KB and CL strategies, Zhan et al. [34] used a fake function to replace the actual EI function and cycle q times, named Pseudo EI (PEI) strategy. PEI infilling strategy has a fast convergence speed in constructing a good kriging model. As EI function tends to converge, the fixation of q makes it easier to generate meaningless points, which increases the cost of optimization.

EI function changes with EGO iteration, and sequential select q multi-point often leads to additional optimization burdens by falling into aggregation. Xie et al. [35] removed duplicate points by calculating candidate point correlations. Gobert et al. [36] proposed an acquisition process technique based on design space partitioning to select valuable candidate points from kriging model. Since EI function has a closed-form expression, the computational cost is negligible compared to the optimization evaluation [27]. Researchers devise adaptive parallel infilling strategies to reduce the optimization burden by sacrificing EI function calculation. Xiao et al. [3] focused on selecting global multi-point and used a refined sampling/importance resampling to search the points with large EI values. Moreover, the of points needs to be set in advance. Zhan et al. [37] constructed EI multi-peak using Latin hypercube sampling and dynamic generation of points by each non-contiguous sub-domain. However, EI function multi-peak selection judgments not be discussed much. Selecting low-information peaks may not be helpful during the optimization process.

For an excellent infilling strategy, points generated should be global and dynamic while avoiding clustering with each other. The difficulty lies in capturing and selecting EI multi-peak for each EGO iteration. Population Monte Carlo (PMC) method has the potential to solve these problems. PMC is an unbiased sampling method consisting of iteratively generated importance samples from global space. Beaumont et al. [38] extended the sampling method to Bayesian inference for approximating target distribution $P(\theta \mid \mathbf{X})$ when likelihood function $P(\mathbf{X} \mid \theta)$ hardly computed or even unknown. As defined in PMC, given a series of decreasing sampling thresholds $\epsilon_1 > \epsilon_2 > \cdots > \epsilon_\ell$ (ℓ being the final iteration) in advance, the intermediate samples are sequentially updated in order to approximate true distribution. The sampling threshold controls range of samples produced and is a key to PMC generating samples closer to target multi-peak distribution.

$$P(\theta \mid \mathbf{X}) \propto P(\mathbf{X} \mid \theta) \cdot P(\theta) \tag{1}$$

PMC generates approximate samples to construct unknown distribution. For converging samples and selecting global candidate points, density-based spatial clustering (DBSCAN) approach is simple and effective. DBSCAN method enables dynamic partition and reorganize samples with density properties. The intuition for clustering is to divide the samples as sub-domains that satisfy the minimum density. Moreover, these sub-domains are constrained by a minimum distance threshold to ensure the diversity of information [39]. Samples in same distribution or adjacent to each other are considered to be in same cluster. In contrast, samples far apart from each other are considered in different distributions, which effectively solves the problem of final candidate points falling into aggregation due to proximity.

This paper proposes an adaptive parallel sampling scheme based on EI infilling strategy, called the parallel Non-parametric Population Monte Carlo sampling scheme (NPMS). The NPMS scheme follows a native idea: taking EI function as a probability density function with a multi-peak form. And in order to reduce parallel strategy burden caused by points aggregation, NPMS scheme obtaines multiple candidate points in staged manner. In the first stage, PMC method samples from global input space and converges samples to individual high EI value sub-domains. A non-parametric sampling threshold method is used to determine whether the areas are selected and save more sampling costs. In the second stage, the neighborhood distance threshold and the sample size threshold are adjusted according to sample density and EI convergence information, and then DBSCAN clustering is performed. The candidate points come from PMC samples with the best EI value in each cluster. Above, NPMS scheme uses the currently known information and makes adaptive decisions by sampling EI function more in return for a lower optimization casts. At the same time, candidate points from interest sub-domains and far from each other, which improving optimization benefits.

The rest of this paper is organized as follows. Section II details the NPMS scheme framework. Section III validates the characteristics and feasibility of the NPMS scheme through a benchmark case. The effect of the parameters on the model is then analyzed. Section IV provides a detailed performance comparison and analysis of the NPMS strategy with other infilling strategies (i.e., EI, GO-HI, CL, KB, and PEI) through benchmark function tests. Section V summarizes the conclusion and future work.

II. NON-PARAMETRIC POPULATION MONTE CARLO SAMPLING SCHEME (NPMS)

This section details the Non-parametric Population Monte Carlo sampling (NPMS) scheme. After kriging model is constructed, NPMS scheme acts on EGO candidate selection process. Defining EI function as a probability density function, NPMS scheme uses two stages: non-parametric population Monte Carlo sampling and density clustering. The flowchart of NPMS applied to EGO as shown in Figure 1.

In the paper, we assume that EGO is applied in continuous space, and the goal is to find a point with the minimum value of unknown objective function f:

$$x^{-} = \arg\min_{x\in\mathbb{X}} f(x) \tag{2}$$

where $\mathbb{X}$ denotes input space of x. For convenience, the obtained EI values are converted into negative numbers

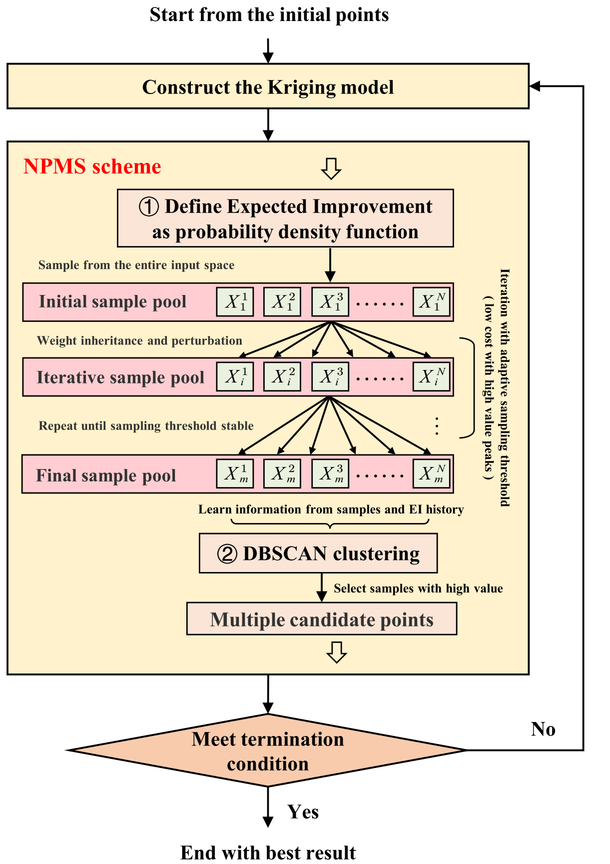


FIGURE 1. The flow chart of NPMS scheme in EGO framework.

shown in Eq.(3).

$$\mathrm{EI}(x) = \begin{cases} -\left[(y_{\min} - \hat{y})\ \Phi\left(\frac{y_{\min}-\hat{y}}{\hat{\sigma}}\right) \right. \\ \qquad \left. +\hat{\sigma}\phi\left(\frac{y_{\min}-\hat{y}}{\hat{\sigma}}\right)\right], & \hat{\sigma} > 0 \\ 0, & \hat{\sigma}(x) = 0 \end{cases} \tag{3}$$

where $\hat{y}$ is kriging prediction value and $\hat{\sigma}$ is corresponding standard deviation. The Φ and ϕ are the cumulative density function and probability density function of a normal distribution, respectively. The multi-peak in EI function represents the minimum value regions.

A. EI MULTI-PEAK SAMPLING

Define multi-peak sub-domains in EI function as the $P(\mathbf{x})$. Samples $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ are obtained by non-parametric method sampling from high EI value sub-domains and used to construct an approximate distribution of EI function $P(\mathbf{x} \mid \mathbf{X})$. For each EGO iteration step, sampling method and sampling threshold selection are performed by following two steps:

Step 1: PMC Sampling Framework Construction:

PMC method initially begins with sufficient statistics. At the initial step $\ell = 1$, Latin Hypercube Sampling (LHS) method is used to generate a set of N global sample pool $\boldsymbol{X}_1 = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$, with corresponding EI value $EI(\boldsymbol{X}_1)$.

For each sample pool, reject the ineligible samples based on the approximate constraint rule Eq.(4), and form prior distribution $P(\boldsymbol{x})$ from the accepted samples.

$$P(\boldsymbol{x} \mid \boldsymbol{X}) \approx P(\boldsymbol{x} \mid EI(\boldsymbol{X}_\ell) \leqslant \epsilon_\ell) \tag{4}$$

where ϵ_ℓ is defined as PMC sampling threshold in range $\min(\mathrm{EI}) \leq \epsilon_\ell \leq 0$. For approximate constraint rule, samples with the EI value greater than ϵ_ℓ will be rejected, while samples in high EI values (less than ϵ_ℓ) are accepted. Use the prior distribution $P(\boldsymbol{x})$ to complete the sample pool $\boldsymbol{X}_1$ to number N, and give initial sample pool weight W_1.

$$W_1 = \left\{ w_1^i \mid w_1^i = \frac{1}{N}, i = 1, \ldots, N \right\} \tag{5}$$

In arbitrary step ($\ell > 1$) in PMC sampling iteration, iterative sample pool $\boldsymbol{X}_\ell$ is derived from $\boldsymbol{X}_{\ell-1}$ via a particle filter methodology under approximate constraints ϵ_ℓ. Specifically, a particle $\boldsymbol{x}^i$ is randomly samples from $\boldsymbol{X}_{\ell-1}$ with weights $\left(W^i_{(\ell-1)}\right)_{i=1,\ldots,N}$ and the new sample $\boldsymbol{x}^*_\ell$ samples from an adaptive Gaussian Markov transition kernel $q\left(\boldsymbol{x}^*_\ell \mid \boldsymbol{x}^i\right)$. In Gaussian Markov transition kernel, the mean is $\boldsymbol{x}^i$ and the variance σ^2 is twice the weighted empirical variance of sample pool $\boldsymbol{X}_{\ell-1}$ as shown in Eq.(6). The weights $\left(W^i_{(\ell)}\right)_{i=1,\ldots,N}$ on each new sample $\boldsymbol{x}^*_\ell \in \boldsymbol{X}_\ell$ assigned in the previous iteration is shown in Eq.(7). As sampling threshold ϵ_ℓ shrinks, the distribution constructed by final sampling pool $\boldsymbol{X}_\ell$ accurately approximates current EI function.

$$\boldsymbol{x}^*_\ell \sim q\left(\boldsymbol{x}^*_\ell \mid \boldsymbol{x}_\ell\right) = \mathcal{N}\left(\boldsymbol{x}_\ell, \sigma^2_{\ell-1}\right) \tag{6}$$

$$w^i_\ell = \frac{P\left(\boldsymbol{x}^i_\ell\right)}{\sum_{j=1}^N W^j_{\ell-1} q\left(\boldsymbol{x}^j_{\ell-1} \mid \boldsymbol{x}^i_\ell, \sigma_{\ell-1}\right)} \tag{7}$$

Step 2: Non-Parametric Sampling Threshold Selection: During iterative sampling, all coefficients are adaptive except sampling threshold ϵ_ℓ. Sampling threshold directly affects cost and accuracy in PMC: (i) a strict threshold allows samples to be distributed in fewer areas with higher EI values, accompanied by high rejection sample sizes and sampling times, (ii) a loose threshold preserves a large number of samples and shortens sampling time in PMC, but inevitably contains meaningless samples. Suppose that at any iteration step ℓ with sampling threshold ϵ_ℓ, another sample pool with N_p ($N < N_p$) samples is sampled from the $\Theta_{\ell-1}$ (using the LHS method in $\ell = 1$). Based on approximate constraint rule in Eq.(4), the number of accepted samples is N_{acc}. The acceptable ratio P_{accept} is defined as:

$$P_{accept} = \frac{N_{accept}}{N_p} \tag{8}$$

For accepted N actual samples after using approximate constraint rule, the required accumulated sample size N_{need} is calculated by acceptable ratio and N as shown in Eq.(9). PMC method should balance sampling cost and information gain limit in multi-peak areas in each sampling iteration, which means the sampling threshold ϵ_ℓ and the number total of samples N_{need} should both be minimum.

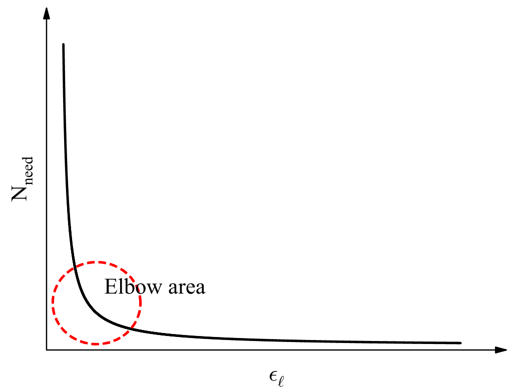


FIGURE 2. For each fixed EI function, N_{need} is inversely proportional to ϵ_ℓ and will not be changed. The red circle is the elbow position of the relationship.

$$N_{need} = \frac{N}{P_{accept}} = \frac{N \times N_p}{N_{accept}} \propto \frac{1}{\epsilon_\ell} \tag{9}$$

We propose to minimize the Euclidean distance from $\epsilon_\ell - N_{need}$ relationship curve to the origin as sampling threshold location, as shown in Figure 2 elbow area. When sampling threshold does not stable, each sampling iteration ℓ is performed according to following steps:

(1) Predefined another sample pool with N_p sample size, $\mathbf{X}_{N_p} = \left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_p}\right\}$. The sample pool is sampled by PMC method without approximate constraint rule in Eq.(4), where each sample based on Eq.(6) and Eq.(7) from the previous generation sample pool $\mathbf{X}_{\ell-1}$. The sampling threshold values $EI(\mathbf{X}_{N_p})$ correspond to each sample calculated by the EI function. Sort samples according to the $EI(\mathbf{X}_{N_p})$ from small to large. Let the sorting indexes $I = 1, \ldots, N_p$ be N_{accept} values.

(2) N_{need} range is $[N, N \times N_p]$ obtained by Eq.(9). The curve between $EI(\mathbf{X}_{N_p})$-N_{need} is inversely proportional as discussed above. Due to the large data difference between the EI value and N_{need}, the logarithmic processing of N_{need} is defined as N_{Lneed}. Cost and accuracy have same status in the balance. Normalised $EI(\mathbf{X}_{N_p})$ and the N_{Lneed} by Eq.(10) (defined as $Nor(EI(\mathbf{X}_{N_p}))$ and $Nor(N_{Lneed})$).

$$Nor(x) = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{10}$$

(3) Final sampling threshold is in elbow area of the $Nor(EI(\mathbf{X}_{N_p}))$ - $Nor(N_{Lneed})$ relationship. Calculate Euclidean distance to the origin for each sample in the relationship curve. And the sampling threshold position corresponds to the minimum distance, with threshold value ϵ_ℓ corresponds to the EI value.

$$\begin{cases} \min & (Nor(N_{Lneed}))^2 + Nor(E_\ell)^2 \\ \text{s.t.} & f\left(Nor(N_{Lneed}), \varepsilon_t\right) = 0 \end{cases} \tag{11}$$

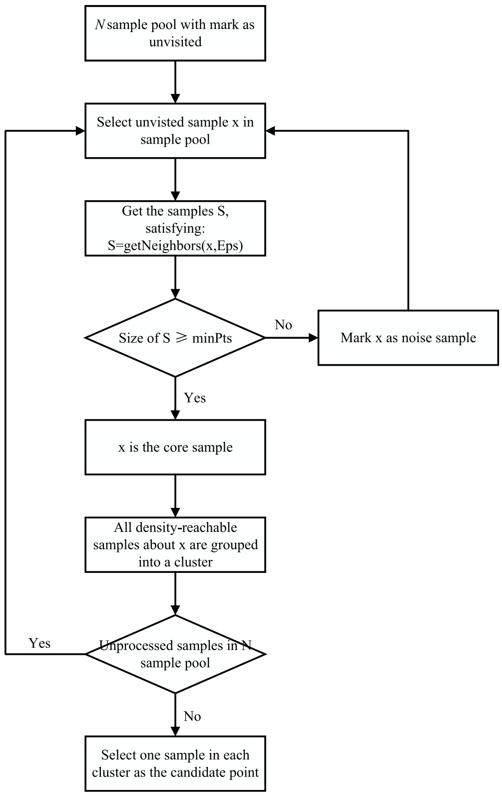


FIGURE 3. DBSCAN method flowchart. The input data is final sample pool generated by NPMS first stage. And the output data are candidate points.

(4) Samples with EI values greater than $epsilon_\ell$ in N_p sample pool are rejected. And replenish the number of accepted samples to N as true iterative sample pool.

B. DENSITY CLUSTERING

Non-parametric PMC sampling method finally generates samples with EI multi-peak characteristic distribution. The DBSCAN method redistributes the final sample pool $\mathbf{X}$: (1) Define the minimum neighborhood distance threshold Eps and the minimum sample size threshold $minPts$, (2) Select each sample in sample pool $\mathbf{X}$, according to Eps and $minPts$ to determine whether it is a core point, if it is a core point, select all vertices whose density is reachable to form a cluster, (3) Find all clusters in $\mathbf{X}$ where the density of all core samples are reachable, and the samples that are not assigned to any cluster are samples as noise points (4) Select samples in each cluster with the minimum EI value as final candidate points. The workflow is shown in Figure 3.

Not each peak in EI function corresponds to a cluster, and emphasis should be placed on the diversity of information for candidate points. The minimum neighborhood distance threshold Eps and the minimum sample size threshold $minPts$ constrain the clustering behavior. Lower Eps and $minPts$ correspond to more candidate points selection. On the contrary, there will be a single-choice phenomenon. NPMS scheme proposes adaptive Eps and $minPts$. The Eps is shown as follows:

$$Eps = \gamma \times \sigma(d(\mathbf{X})) \tag{12}$$

Algorithm 1 The Framework of NPMS Scheme

Input: Constructed kriging model, N_p, N, γ, β
Output: Candidate points $\mathbf{X_{next}}$

1: Sampling initial pool $\mathbf{X}_{N_p}$ from LHS method, and sorting by EI value
2: Establishing the relationship between the N_{Lneed} and $EI(\mathbf{X}_{N_p})$ using Eq.(9) and Eq.(10)
3: Calculating the sampling threshold ϵ_1 with the minimum distance
4: Retaining the samples using Eq.(4), and getting the prior distribution $P(\mathbf{x})$
5: Completing $\mathbf{X}_{N_p}$ to N as first sample pool $\mathbf{X}_1$
6: Setting weight $W_1 = \frac{1}{N}, \ldots, \frac{1}{N}$
7: **while** sampling threshold does not stable **do**
8: Sampling $\mathbf{X}_{N_p}$ from $\mathbf{X}_{\ell-1}$ by PMC method using Eq.(6)
9: Getting the sampling threshold ϵ_ℓ using step 2-3
10: Completing the iterative sample pool $\mathbf{X}_\ell$ to number N
11: Setting $\sigma_\ell^2 \leftarrow 2Cov(\Theta_\ell)$
12: Setting W_ℓ through Eq.(7)
13: **end while**
14: Calculating the cluster threshold Eps and $minPts$ by Eq.(12) and Eq.(13)
15: Clustering the final sample pool $\mathbf{X}$
16: Selecting the minimum EI value in each cluster as final candidate points $\mathbf{X_{next}}$

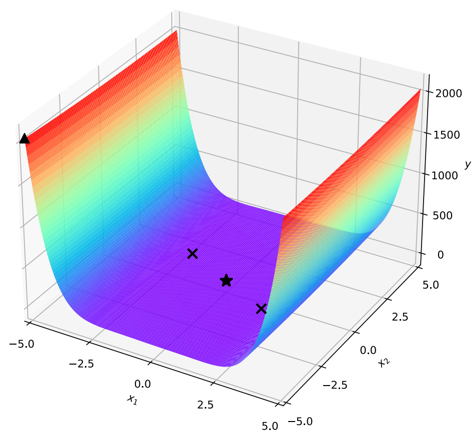


FIGURE 4. Contour plot of the Camel3 case, where "X" are two local minimums. The global minima and the maximum corresponding to the asterisk and triangle.

where $d(\cdot)$ represents the distance from each sample to the origin, σ is the standard deviation of $\mathbf{X}$. Eps defines a distance threshold based on overall sample dispersion, regulated by γ parameter. $minPts$ varies as EI converges and is defined as follows:

$$minPts = \left\lfloor N \times \frac{\beta}{1 + \exp\left(\left|\frac{\epsilon}{\min(eis)}\right|\right)} \right\rfloor \tag{13}$$

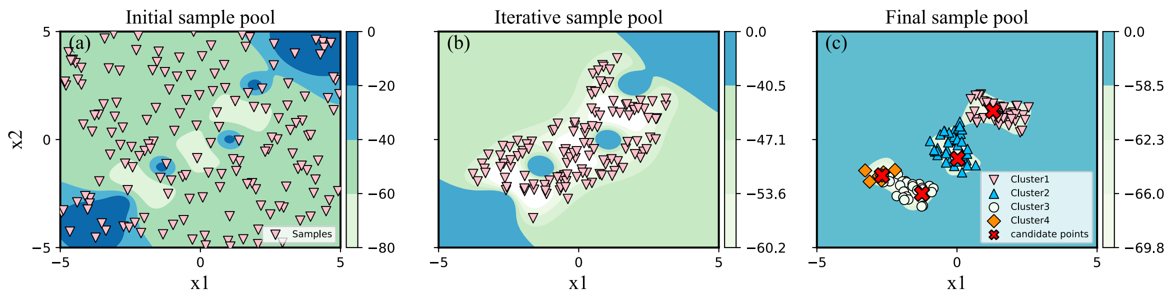


FIGURE 5. NPMS scheme sampling process at the 1^{st} EGO iteration, where the background contour is the EI function.

where ϵ is the final sampling threshold value in NPMS. $\min(eis)$ represents the minimum sampling threshold of EGO iteration history. In the end, NPMS scheme pseudocode can be expressed as Algorithm.1.

III. BENCHMARK

In this section, a benchmark case is used to analyze the properties of NPMS scheme. First, we validate characteristics of scheme itself and its performance in EGO applications. And then compare it with the original EI strategy to illustrate the feasibility and advantages. Finally, model parameters' effect is analyzed, and reasonable values are given.

The benchmark is named Three-hump camel-back (Camel3) function, and formula is shown in Eq.(14). Camel3 has a regular U-shaped structure in space shown in Figure 4, with one global minimum, two local minimums, and one global maximum. The difficulty of capturing changes in function values around global and local minima makes optimization take a long to converge.

$$f(\mathbf{x}) = 2x_1^2 - 1.05\, x_1^4 + \frac{x_1^6}{6} + x_1x_2 + x_2^2, x_i \in [-5, 5] \quad (14)$$

Five indicators are involved in quantifying the impact of strategy. The standardized average LOO CV (CV) and coefficient of determination (R^2) [40] are used to evaluate the prediction performance of kriging model:

$$CV = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}}{\text{range of } y_1, \ldots, y_i} \quad (15)$$

$$\mathrm{R}^2 = 1 - \frac{\sum_{i=1}^{T}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{T}\left(y_i - \bar{y}\right)^2} \quad (16)$$

where n and T are the numbers of observable points and test points. y_i and $\hat{y}_i$ denote the true responses and predicted responses. $\bar{y}$ is the mean of true responses. $CV < 0.1$ is used as the EGO stopping condition in the benchmark.

To evaluate the performance of NPMS applied in EGO, I_s, N_s, and Res are proposed. I_s is the total number of EGO sequential iterations and also measures the speed of EGO optimization convergence. N_s is the total number of candidate points required by infilling strategy when EGO meets stopping condition. N_s determines the computational cost optimization. Res is defined as residual of global minimum and optimal result, with more accurate optimization results implying a smaller Res. For the above metrics, higher R^2 and lower CV, I_s, N_s, Res are usually desired. During benchmark, all initial points [41] and test points [25] are generated by the LHS method following the maximin criterion. And all benchmarks are independently repeated 30 times.

A. CONVERGENCE AND ADAPTIVE

First, we verify the convergence of the NPMS scheme itself. In the non-parametric PMC sampling stage, the NPMS scheme aims to converge the sample pool to high EI value sub-domains. The initial NPMS scheme generates a uniform sample pool containing multi-peak information. As sampling iterative, samples inherit from the previous sample pool and converge to target multi-peak constrained by sampling threshold, as shown in Figure 5 (b). Until sampling threshold becomes stable, PMC sampling process is complete. Samples are finally distributed among the EI function multi-peak from entire space, while the areas with EI values smaller than sampling threshold are considered meaningless. NPMS aims to converge the samples to candidate points during clustering stage. Based on density features, the sample pool is re-divided into new clusters, ensuring each cluster adequately expresses information gain of EI function. The candidate points are selected with the best EI values in each new cluster, as shown in Figure (c) red points. Non-parametric sampling can sample from high EI value areas, while clustering can select global points. Thus, the NPMS scheme itself is convergent.

When applied to EGO, NPMS scheme has good sampling threshold convergence and dynamics candidate points selection. For each EGO iteration, the fixed EI function ensures that sampling threshold relationship in NPMS scheme does not change. Furthermore, the actual sampling threshold position will not change either. Due to sampling uncertainty, the threshold will fluctuate during sampling iterative and tend to be stable. Figure 6 shows the history of sampling threshold in NPMS scheme when at 20^{th}, 40^{th} and 60^{th} EGO iterations, respectively. Each NPMS scheme can obtain a stable final sampling threshold. And as EGO iteration proceeds, sampling threshold tends to zero as shown in Figure 6 (d). Secondly, adaptive clustering based on sample density ensures NPMS scheme dynamics candidate points selection. With the requirement that each cluster maintains

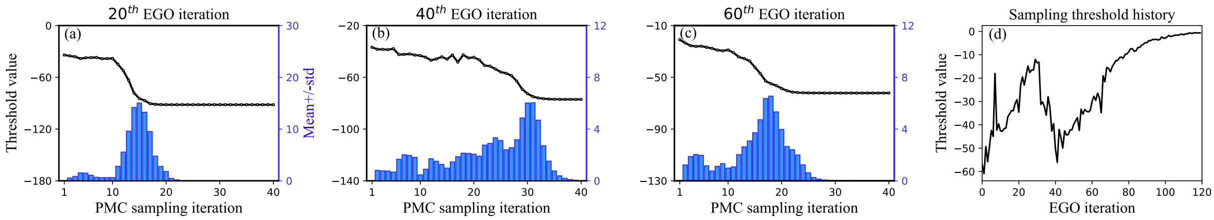


FIGURE 6. **The sampling threshold selection history. Figure (a), (b) and (c) represent the sampling thresholds history generated in 40 times NPMS sampling iterations. Each figure corresponds to one EGO iteration. The black lines are the sampling threshold value, and the blue bars are the statistical standard deviation. Figure (d) is the history of final sampling threshold generated in EGO iteration.**

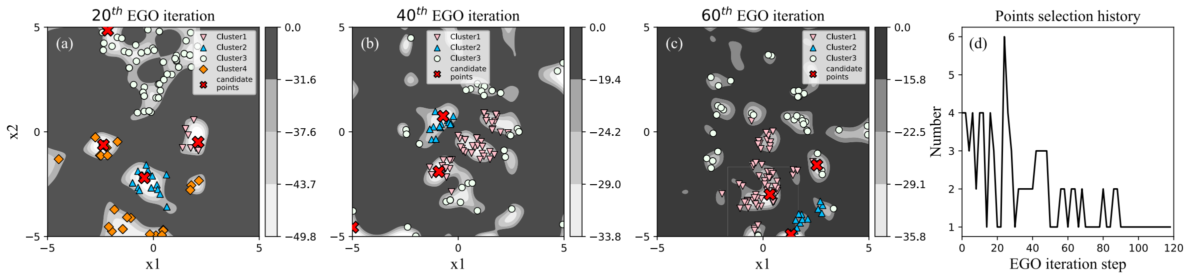


FIGURE 7. **Sample clustering and the number of candidate points selected. Figure (a), (b) and (c) are the clustering results and candidate points of NPMS scheme at EGO iteration in 20^{th}, 40^{th} and 60^{th}, where samples of different colors belong to different clusters. (d) is NPMS scheme candidate point selected number in each EGO iteration.**

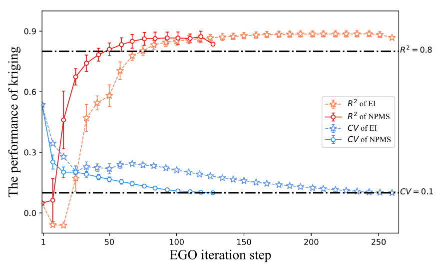


FIGURE 8. **The CV and R^2 convergence history (mean ± std) for EI strategy and NPMS scheme.**

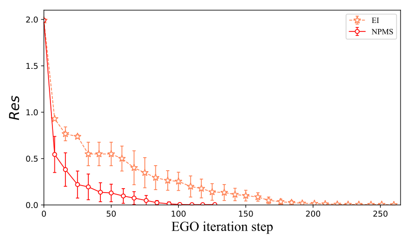


FIGURE 9. **The *Res* convergence history (mean ± std) for EI strategy and NPMS scheme.**

its distance, avoiding the problem of selected candidates falling into aggregation. Figure 7 shows the NPMS scheme clustering and points selection in the EGO iteration. Samples come from global sub-domains with high EI values, and the candidate points are derived from these promising samples, which effective collaboration in sampling and clustering. As shown in Figure 7 (d), NPMS scheme maintains the candidate point selection dynamics when EI function changes and tends to single-point as EI function convergence.

To illustrate the performance of NPEI scheme, we compare it with original EI strategy. Figure 8 and Figure 9 show the convergence histories of EI strategy and NPMS scheme for the Camel3 case. Benchmark is conducted in 30 independent trials, and the average results are shown in Table 1, where the best results are marked in bold. Result shows that samples selected by the two strategies can construct useful models. In the iterative history of EGO, NPMS scheme is faster in building kriging models in the initial steps and has a significant tendency to converge in finding the optimal values. The performance improvement of EGO comes from the broader range of EI function calls and efficient multi-point selection in NPMS scheme. In the first stage, the PMC method generates more samples and performs EI calculations during the iterative process, resulting in the sample size that can be selected being much larger than that of the original EI strategy. The average time consumed internally in each NPMS scheme increased by 77.1% compared to the EI strategy. A large number of samples is more likely to contain optimal values, which results in a residual reduction of 14.6% compared to EI strategy. Meanwhile, the clustering in NPMS second stage can efficiently filter out representative candidate points, reducing 15.8% candidate point number compared to EI strategy. The cost of computing candidate points in EGO is often much higher than calling EI functions, and it is worthwhile for NPMS solutions to improve optimization results and reduce the number of evaluations by calling more EI functions.

For demonstrating global and efficient nature of NPMS, the candidate points distribution of EI and NPMS is analyzed, as shown in Figure 10 (a) and (b). The objective function

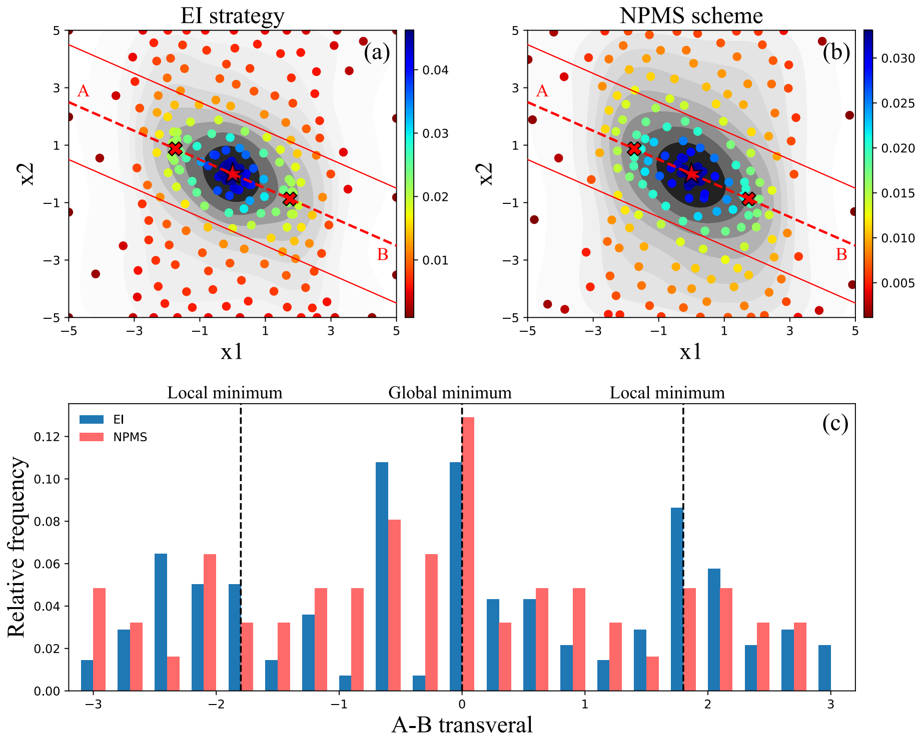


FIGURE 10. **(a) Contour plot of total candidate points generated by EI strategy, where the red dashed line connects the global minimum and two local minima of the objective function. Each solid red line differs from the dashed line by a distance of 2. (b) Contour plot of total candidate points generated by NPMS scheme. (c) Relative frequency of points in EI strategy and NPMS scheme on the A-B transversal, where the transversal range focuses on convergence area.**

TABLE 1. **The results of EI strategy and NPMS scheme in Camel3 case.**

Strategy	Res	N_s	I_s	CV	R^2
EI	0.0041	253.6	247.6	0.0997	**0.886**
NPMS	**0.0035**	**213.7**	**110.2**	**0.0996**	0.864

has one global minimum and two local minima on the same line A-B, as shown in red dashed line. Both EI and NPMS strategies can converge to a global minimum during optimization. However, EI strategy is prone to falling into local optima and tends to fall into a state of point aggregation. In contrast, NPMS scheme is more decentralized in the objective function space. To better represent the distribution, candidate points within selected range (solid lines) are projected onto dashed line. Their distances to the global minimum are counted, yielding the statistics shown in Figure 10 (c). EI strategy produces a fluctuation distribution of candidate points and often higher relative frequencies of points in the two local minima regions. In contrast, NPMS distribution is more uniform, with no significant low peaks, while focusing more on the global minimum area.

B. MODEL PARAMETER ANALYSIS

The NPMS scheme involves four parameters in optimization process: N, N_p, γ, and β. N and N_p are sampling parameters of NPMS first stage, representing the sampling number in actual sample pool and pre-sample pool, respectively. N_p involves generating a non-parametric sampling threshold. N significantly influences the NPMS scheme, which determines sampling cost and candidate selection accuracy. Large sample sizes can be stably distributed over multi-peak regions, resulting in a high sampling cost. Small sample size can maintain a low sampling cost but cannot accurately capture the multiple peaks of the EI function, resulting in an unstable candidate selection capability. γ and β are the clustering parameters, which influence the ability of the NPMS scheme to balance informativeness and diversity at candidate points. High γ and β can cause NPMS scheme to fail to select enough multiple points and degrade to a single-point strategy. Conversely, low γ and β will cause NPMS scheme to produce aggregated candidate points, resulting in redundancy. As mentioned above, since scheme does not require the highest sampling accuracy, setting $N_p = 200d$ as a recommended reference value. N sets to the ratio of N_p, taking the range in [0, 1]. The range of γ and β are in [0,1].

Figure 11 shows mean and standard deviation of the optimization results Res and the optimization iterations I_s (N_s has no significant difference) under parameter N in [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9] where γ and β are fixed as a median value 0.5. For optimization result Res, the mean and standard deviation fluctuate as N rises. The mean and

TABLE 2. The optimization results and rankings in different combinations of parameters.

β	γ	Res (ranking)	N_s (ranking)	I_s (ranking)	Average rank
0.1	0.1	0.0045 ± 0.0055 (7)	254.0000 ± 26.8663 (24)	64.1000 ± 12.3730 (1)	10.7 (7)
0.1	0.3	0.0127 ± 0.0111 (17)	207.7000 ± 12.4503 (11)	140.7500 ± 12.7784 (19)	15.7 (21)
0.1	0.5	0.0304 ± 0.0253 (24)	203.9500 ± 13.5553 (9)	136.3500 ± 13.3427 (16)	16.3 (23)
0.1	0.7	0.0340 ± 0.0362 (25)	209.4000 ± 13.4253 (14)	142.6500 ± 14.0188 (20)	19.7 (25)
0.1	0.9	0.0191 ± 0.0207 (21)	206.1000 ± 11.7979 (10)	139.3000 ± 12.4222 (18)	16.3 (24)
0.3	0.1	0.0017 ± 0.0018 (1)	257.4500 ± 20.5048 (25)	70.9500 ± 7.7361 (2)	9.3 (2)
0.3	0.3	0.0024 ± 0.0022 (2)	226.3500 ± 19.1214 (22)	110.7000 ± 10.3976 (9)	11.0 (10)
0.3	0.5	0.0102 ± 0.0067 (16)	182.9000 ± 10.4014 (5)	163.9000 ± 9.4175 (23)	14.7 (18)
0.3	0.7	0.0187 ± 0.0246 (20)	175.2500 ± 9.9693 (3)	167.4000 ± 10.4422 (24)	15.7 (22)
0.3	0.9	0.0185 ± 0.0191 (19)	174.9000 ± 13.9173 (2)	169.8000 ± 13.4818 (25)	15.3 (20)
0.5	0.1	0.0056 ± 0.0075 (10)	213.8500 ± 18.0867 (17)	95.1500 ± 9.4777 (3)	10.0 (4)
0.5	0.3	0.0033 ± 0.0036 (3)	227.3000 ± 17.9697 (23)	103.6500 ± 8.4396 (4)	10.0 (3)
0.5	0.5	0.0036 ± 0.0041 (4)	209.2000 ± 17.2093 (13)	112.2500 ± 8.6884 (11)	9.3 (1)
0.5	0.7	0.0093 ± 0.0141 (15)	194.2500 ± 15.1059 (7)	146.5500 ± 9.9019 (21)	14.3 (17)
0.5	0.9	0.0283 ± 0.0326 (23)	173.6000 ± 8.8848 (1)	159.5000 ± 8.4705 (22)	15.3 (19)
0.7	0.1	0.0064 ± 0.0061 (13)	198.3500 ± 15.7616 (8)	115.9500 ± 10.3705 (12)	11.0 (8)
0.7	0.3	0.0041 ± 0.0050 (5)	212.0500 ± 19.2366 (16)	110.9500 ± 10.9292 (10)	10.3 (5)
0.7	0.5	0.0060 ± 0.0094 (11)	215.8500 ± 19.9581 (19)	107.5500 ± 9.9523 (5)	11.7 (11)
0.7	0.7	0.0072 ± 0.0095 (14)	218.4500 ± 18.1727 (21)	108.3500 ± 9.6968 (6)	13.7 (15)
0.7	0.9	0.0053 ± 0.0054 (9)	211.7500 ± 15.498 (15)	126.8000 ± 7.9787 (15)	13.0 (13)
0.9	0.1	0.0217 ± 0.0287 (22)	181.3500 ± 16.6231 (4)	137.1000 ± 14.4323 (17)	14.3 (16)
0.9	0.3	0.0158 ± 0.0240 (18)	189.4000 ± 20.2322 (6)	121.9000 ± 14.3698 (14)	12.7 (12)
0.9	0.5	0.0049 ± 0.0068 (8)	208.0500 ± 13.4888 (12)	116.8000 ± 6.4545 (13)	11.0 (9)
0.9	0.7	0.0042 ± 0.0041 (6)	214.6500 ± 23.6332 (18)	110.4500 ± 12.8354 (8)	10.7 (6)
0.9	0.9	0.0061 ± 0.0058 (12)	217.6000 ± 17.2812 (20)	109.2000 ± 8.4593 (7)	13.0 (14)

TABLE 3. Benchmark cases functions.

Group	Dimension	Function	Formulation	Range
group 1	2	Leon (Leo)	$f(\mathbf{x}) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$	$x_i \in [-1.2, 1.2]$
	2	Alpine 2 (Alp)	$f(\mathbf{x}) = -\sqrt{x_1 x_2}\sin(x_1)\sin(x_2)$	$x_i \in [0, 10]$
	2	Bukin 6 (Buk)	$f(\mathbf{x}) = 100\sqrt{\left\lvert x_2 - 0.01x_1^2\right\rvert} + 0.01\lvert x_1 + 10\rvert$	$x_1 \in [-15, -5], x_2 \in [-3, 3]$
group 2	2	Cube (Cub)	$f(\mathbf{x}) = 100\left(x_2 - x_1^3\right)^2 + (1 - x_1)^2$	$x_i \in [-10, 10]$
	2	Holder table 2 (Hol)	$f(\mathbf{x}) = -\left\lvert \sin(x_1)\cos(x_2)\exp\left(\left\lvert 1 - \left(x_1^2 + x_2^2\right)^{0.5}/\pi\right\rvert\right)\right\rvert$	$x_i \in [-10, 10]$
	2	Cross-in tray (Cro)	$f(\mathbf{x}) = -0.0001\left(\left\lvert \sin(x_1)\sin(x_2)\exp\left(\left\lvert 100 - \left(x_1^2 + x_2^2\right)^{0.5}/\pi\right\rvert\right)\right\rvert + 1\right)^{0.1}$	$x_i \in [-10, 10]$
group 3	2	El-Attar-Vidyasagar-Dutta (EAVD)	$f(\mathbf{x}) = \left(x_1^2 + x_2 - 10\right)^2 + \left(x_1 + x_2^2 - 7\right)^2 + \left(x_1^2 + x_2^3 - 1\right)^2$	$x_i \in [-500, 500]$
	2	Bohachevsky 1 (Boh)	$f(\mathrm{x}) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$	$x_i \in [-100, 100]$
	2	Bartels Conn (Bar)	$f(\mathbf{x}) = \left\lvert x_1^2 + x_2^2 + x_1 x_2\right\rvert + \lvert\sin(x_1)\rvert + \lvert\cos(x_2)\rvert$	$x_i \in [-500, 500]$
group 4	4	DIXON-PRICE (Dix)	$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^{4} i\left(2x_i^2 - x_{i-1}\right)^2$	$x_i \in [-10, 10]$
	6	Hartmann 6 (Hart)	$f(\mathbf{x}) = -\sum_{i=1}^{4}\alpha_i\exp\left(-\sum_{j=1}^{6}A_{ij}\left(x_j - P_{ij}\right)^2\right)$	$x_i \in [0, 1]$
	8	Trid Function 8 (Trid)	$f(\mathbf{x}) = \sum_{i=1}^{8}(x_i - 1)^2 - \sum_{i=2}^{8}x_i x_{i-1}$	$x_i \in [-64, 64]$
	10	Levy Function 10 (Levy)	$f(\mathrm{x}) = \sin^2(\pi\omega_1) + \sum_{i=1}^{9}(\omega_i - 1)^2\left[1 + 10\sin^2(\pi\omega_i + 1)\right] + (\omega_{10} - 1)^2\left[1 + \sin^2(2\pi\omega_{10})\right]$	$x_i \in [-10, 10]$

standard deviation of *Res* decrease as N increases until N is 0.3. When N is greater than 0.5, NPMS scheme could find a good optimization result, but the average of *Res* tends to increase. In contrast, N is more optimally stable between 0.3 and 0.5. The mean value of total optimization iterations I_s tends to increase as N increases, while the standard deviation of I_s does not change significantly during the change. In detail, the iteration I_s increases slightly between $N = 0.3$ and $N = 0.4$. Overall, $N = 0.3$ is a good choice for the trade-off between optimization results and iteration speed.

Table 2 shows the results of γ and β in different combinations, and values after "$\pm$" represent the standard deviations. Average rank is calculated by *Res*, I_s, and N_s. Parameters are taken at 0.2 intervals, and N is fixed at the suggested value. The γ and β have synergistic effects. When both γ and β are less than 0.3, NPMS scheme had fewer EGO iterations but produced more candidate points. When both γ and β are greater than 0.5, EGO results are stable, but I_s and N_s tend to rise. NPMS scheme loses advantage of adaptive multi-point, resulting in poorer rankings. When a parameter is too small, and the other is dominant, too large value choices can lead to poor I_s and N_S and unstable optimization results. It is difficult to make a set of parameters that will give NPMS the best I_s, N_s, and *Res* in the optimization process. Based on

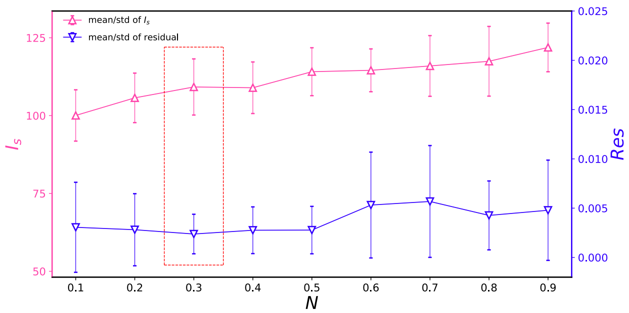


FIGURE 11. The required I_s and Res under the different values of N.

the average ranking, $\gamma = 0.5$ and $\beta = 0.5$ are reasonable choices and then applied in the subsequent sections.

IV. PERFORMANCE COMPARISON WITH OTHER STRATEGIES

This section compares NPMS with two single-point infilling strategies (i.e., EI and GO-HI) and three parallel strategies (i.e., CL, KB, and PEI) under four test groups. NPMS scheme uses the parameters discussed above. The tests are conducted independently 30 times, with kriging reaching $CV < 0.1$ first time as EGO convergence condition.

The benchmark cases are provided in Table 3. Nine cases of the same dimension are divided into three groups according to the input space size (V), which is defined by its Euclidean volume:

$$V = \prod_{i=1}^{d} (ub_i - lb_i) \tag{17}$$

where ub_i and lb_i denote the input space's upper and lower bound. The input space size is categorized into three classes, small ($V \leq 100$), medium ($100 < V \leq 400$), and large ($V > 400$). And the cases in each group are selected in increasing order of difficulty (modality, separability), as discussed in Jamil and Yang [42].

Group 4 focuses on testing the adaptability of each infilling strategy to dimensional changes, in which DIXON-PRICE, Hartmann, Trid Function, and Levy Function dimensions correspond to 4, 6, 8, and 10, respectively. The α, A and P in Har6 case and ω_i in Levy case is defined as follow:

$$\alpha = (1.0, 1.2, 3.0, 3.2)^T \tag{18}$$

$$A = \begin{pmatrix} 10 & 3 & 17 & 3.50 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix} \tag{19}$$

$$P = 10^{-4} \begin{pmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{pmatrix} \tag{20}$$

$$\omega_i = 1 + \frac{x_i - 1}{4} \tag{21}$$

Table 4 and Table 5 show the performance of kriging model for strategies, where the name of parallel infilling strategies are tilted, and the best results marked in bold. The optimization results (Res, N_s and I_s) for each strategy are shown in Table 6 and Table 7. The Friedman test results are shown in Table 8.

For group 1, due to small input space size for cases, all infilling strategies can build a useful kriging model and find a good minimum at a low computational cost. The parallel infilling strategy requires more candidate point numbers while shorter iterations. In the Leo case, GO-HI infilling strategy shows the lowest candidate point number but needs more iterations than NPMS scheme. The PEI strategy shows the best kriging model performance but has a computational cost close to twice as high as NPMS scheme. In the Alp case, NPMS scheme saves at least 20% of the average optimization computational cost and needs 39.1% of the average number of iterations compared to single-point strategies. NPMS scheme has the lowest computational cost compared to other parallel strategies and has unstable optimization and kriging performance. The NPMS scheme demonstrates a cost-saving advantage in Buk case, which needs only 77% of N_s compared to other strategies but has the worst kriging model. NPMS scheme has a slight advantage over other infilling strategies for the small input size group and has a poor EGO iteration time. Still, it ranks first in candidate point savings and optimization results.

For group 2 (Cub, Hol, Cro), as optimization difficulty increases, strategies require more candidate points and EGO iterations. The fitting ability of constructed kriging model also decreases. In the Cub case, all strategies keep a R^2 score of kriging above 0.9, and NPMS scheme does not get the best results. However, NPMS scheme can find better results in the same iterations compared to other parallel strategies while saving at least 52.5% of the average candidate point number. In the Hol case and Cro case, all strategies do not build a valid kriging model, and complex functions often slow single-point strategies to reach EGO-stopping conditions. While parallel infilling strategies reach EGO-stopping conditions with fewer iterations. However, apart from the NPMS scheme, other parallel strategies required more candidate points than single-point strategies. Compared to single-point strategies, the N_s required for NPMS scheme is reduced by at least 16.8% with similar optimization results found. Compared to CL strategy, NPMS scheme saves at least 13% of the computational cost and improves the results by 41.1%. In the medium space case, NPMS scheme provides excellent result accuracy and cost savings in comparison while requiring more optimization iterations.

For group 3 (Boh, Bar, EAVD), due to the increased search range, the optimal value is not always found accurately in each strategy experiment, but NPMS is shown to have the best average optimization results and minor result fluctuations. Figure 12 and Figure 13 show residual results and convergence history for each infilling strategy. The large input size resulted in infilling strategies failing to construct

TABLE 4. The kriging performance in group 1 and group 2.

Function	Strategy	CV			R^2		
		mean±std	min	max	mean±std	min	max
Leo (d=2)	EI	0.0960 ± 0.0047	0.0844	0.0999	0.9903 ± 0.0039	0.9845	0.9972
	GO-HI	**0.0813** ± 0.0140	0.0554	0.0988	0.9950 ± 0.0023	0.9884	0.9984
	CL	0.0966 ± 0.0045	0.0806	0.1000	0.9953 ± 0.0029	0.9837	0.9991
	KB	0.0911 ± 0.0063	0.0779	0.0982	0.9878 ± 0.0044	0.9746	0.9946
	PEI	0.0977 ± 0.0013	0.0942	0.0998	**0.9988** ± 0.0005	0.9970	0.9992
	NPMS	0.0979 ± 0.0016	0.0937	0.0998	0.9900 ± 0.0017	0.9873	0.9954
Alp (d=2)	EI	0.0985 ± 0.0013	0.0944	0.0999	0.9854 ± 0.0043	0.9780	0.9970
	GO-HI	0.0980 ± 0.0025	0.0881	0.0999	0.9846 ± 0.0058	0.9697	0.9981
	CL	0.0968 ± 0.0034	0.0851	0.1000	0.8344 ± 0.1436	0.5941	0.9694
	KB	**0.0947** ± 0.0047	0.0804	0.0998	0.9672 ± 0.0021	0.9599	0.9696
	PEI	0.0966 ± 0.0031	0.0893	0.1000	0.9658 ± 0.0047	0.9470	0.9695
	NPMS	0.0974 ± 0.0017	0.0942	0.0999	**0.9861** ± 0.0072	0.9696	0.9950
Buk (d=2)	EI	0.0954 ± 0.0057	0.0781	0.0999	0.8279 ± 0.466	-1.5389	0.9931
	GO-HI	0.0952 ± 0.0052	0.0815	0.0999	0.9464 ± 0.0418	0.8312	0.9871
	CL	0.0953 ± 0.0042	0.0823	0.0998	**0.9534** ± 0.0419	0.8363	0.9939
	KB	0.0909 ± 0.0066	0.0755	0.0992	0.9132 ± 0.1230	0.3576	0.9934
	PEI	0.0917 ± 0.0083	0.0611	0.0997	0.9310 ± 0.0999	0.4308	0.9949
	NPMS	**0.0859** ± 0.0163	0.0520	0.0999	0.6736 ± 0.3660	-0.5319	0.9982
Cub (d=2)	EI	0.0996 ± 0.0002	0.0992	0.0999	0.9291 ± 0.0055	0.9160	0.9352
	GO-HI	0.0990 ± 0.0009	0.0958	0.1000	**0.9765** ± 0.0059	0.9653	0.9875
	CL	0.0995 ± 0.0003	0.0992	0.1000	0.9241 ± 0.0068	0.9030	0.9323
	KB	0.0995 ± 0.0003	0.0989	0.1000	0.9146 ± 0.0079	0.8981	0.9292
	PEI	0.0996 ± 0.0003	0.0989	0.1000	0.9229 ± 0.0159	0.8945	0.9592
	NPMS	**0.0989** ± 0.0010	0.0954	0.1000	0.9275 ± 0.0055	0.9164	0.9349
Hol (d=2)	EI	0.0988 ± 0.0018	0.0914	0.1000	0.3648 ± 0.2291	-0.3914	0.8051
	GO-HI	0.0993 ± 0.0006	0.0972	0.1000	0.3430 ± 0.1624	0.1028	0.7955
	CL	0.0986 ± 0.0014	0.0942	0.1000	0.1436 ± 0.3776	-0.8477	0.8019
	KB	**0.0982** ± 0.0020	0.0921	0.0999	0.3650 ± 0.1813	-0.0491	0.8789
	PEI	0.0985 ± 0.0015	0.0931	0.0999	0.3483 ± 0.1292	0.1669	0.7218
	NPMS	0.0990 ± 0.0007	0.0970	0.1000	**0.3790** ± 0.1366	0.0419	0.5496
Cro (d=2)	EI	0.0984 ± 0.0050	0.0722	0.1000	0.3823 ± 0.3153	-0.7719	0.7268
	GO-HI	0.0985 ± 0.0027	0.0867	0.1000	**0.5763** ± 0.1354	0.1701	0.7320
	CL	0.0975 ± 0.0028	0.0885	0.1000	0.2275 ± 0.9430	-4.4112	0.8179
	KB	0.0983 ± 0.0013	0.0936	0.1000	0.5424 ± 0.5930	-2.4865	0.7851
	PEI	0.0981 ± 0.0016	0.0936	0.1000	0.1954 ± 1.0158	-3.7730	0.8358
	NPMS	**0.0960** ± 0.0026	0.0849	0.0982	0.3814 ± 0.2964	-0.3686	0.7386

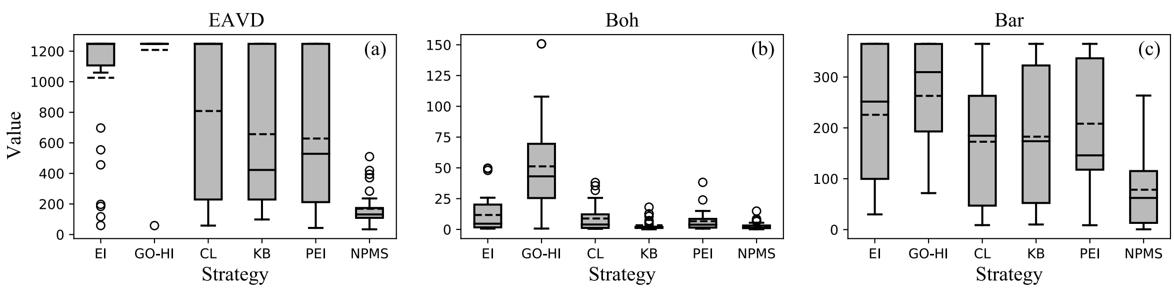


FIGURE 12. The boxplots of the residuals for different strategies in group 3.

kriging models with good predictive power. Each strategy required more candidate points and iterations to sample from the uncertainty sub-domains. The parallel strategy has the advantage of reducing optimization iterations. In

TABLE 5. The kriging performance in group 3 and group 4.

Function	Strategy	CV			R^2		
		mean±std	min	max	mean±std	min	max
EAVD (d=2)	EI	0.0996 ± 0.0002	0.0993	0.1000	0.1581 ± 0.0497	0.0377	0.2367
	GO-HI	0.0996 ± 0.0002	0.0993	0.1000	**0.6134** ± 0.0525	0.5317	0.7173
	CL	0.0997 ± 0.0002	0.0991	0.1000	0.0438 ± 0.026	0.0038	0.0953
	KB	0.0996 ± 0.0002	0.0992	0.1000	0.0378 ± 0.0229	-0.0128	0.0745
	PEI	0.0997 ± 0.0002	0.0992	0.1000	0.0464 ± 0.0235	-0.0101	0.0866
	NPMS	**0.0996** ± 0.0003	0.0991	0.1000	-0.0618 ± 0.0315	-0.1290	-0.0003
Boh (d=2)	EI	0.0998 ± 0.0001	0.0995	0.1000	-1.1693 ± 0.0924	-1.3206	-0.9965
	GO-HI	**0.0996** ± 0.0002	0.0989	0.1000	**0.0738** ± 0.0107	0.0579	0.1003
	CL	0.0997 ± 0.0002	0.0995	0.1000	-1.3213 ± 0.0570	-1.4360	-1.2097
	KB	0.0998 ± 0.0002	0.0995	0.1000	-1.3518 ± 0.0521	-1.4735	-1.2529
	PEI	0.0998 ± 0.0002	0.0995	0.1000	-1.1962 ± 0.0442	-1.2870	-1.1146
	NPMS	0.0997 ± 0.0002	0.0995	0.1000	-0.9085 ± 0.0522	-1.0106	-0.7778
Bar (d=2)	EI	0.0992 ± 0.0030	0.0842	0.1000	-0.0667 ± 0.0302	-0.1471	-0.0103
	GO-HI	0.0992 ± 0.0034	0.0812	0.1000	**-0.0047** ± 0.0071	-0.0221	0.0062
	CL	**0.0990** ± 0.0033	0.0816	0.1000	-0.1273 ± 0.0395	-0.2023	-0.0503
	KB	0.0995 ± 0.0009	0.0959	0.1000	-0.1709 ± 0.0379	-0.2846	-0.0965
	PEI	0.0996 ± 0.0002	0.0992	0.1000	-0.1259 ± 0.0322	-0.1819	-0.0658
	NPMS	0.0997 ± 0.0002	0.0993	0.1000	-0.5083 ± 0.0485	-0.6104	-0.4014
Dix (d=4)	EI	0.0998 ± 0.0001	0.0996	0.1000	-1.2143 ± 0.0405	-1.2867	-1.1368
	GO-HI	0.0997 ± 0.0002	0.0994	0.1000	**-0.5258** ± 0.0389	-0.6024	-0.4428
	CL	**0.0989** ± 0.0006	0.0977	0.0999	-1.1066 ± 0.0609	-1.1825	-0.9505
	KB	0.0990 ± 0.0006	0.0978	0.1000	-1.0672 ± 0.0666	-1.1700	-0.9074
	PEI	0.0994 ± 0.0003	0.0991	0.0999	-0.9028 ± 0.0340	-0.9964	-0.8382
	NPMS	0.0996 ± 0.0002	0.0989	0.0999	-1.0262 ± 0.0795	-1.1904	-0.8600
Hart (d=6)	EI	0.0984 ± 0.0016	0.0924	0.1000	0.1679 ± 0.2257	-0.5601	0.5024
	GO-HI	0.0985 ± 0.0018	0.0917	0.1000	0.5373 ± 0.0692	0.4183	0.6573
	CL	0.0984 ± 0.0013	0.0945	0.1000	**0.6033** ± 0.0635	0.4809	0.7265
	KB	0.0980 ± 0.0017	0.0928	0.0999	0.3839 ± 0.1977	-0.4654	0.6076
	PEI	**0.0979** ± 0.0018	0.0922	0.0999	0.5311 ± 0.0990	0.3380	0.6894
	NPMS	0.0984 ± 0.0020	0.0908	0.1000	-0.0596 ± 0.2861	-0.8196	0.3221
Trid (d=8)	EI	**0.0966** ± 0.0103	0.0474	0.1000	-0.0062 ± 0.0080	-0.0310	0
	GO-HI	0.0974 ± 0.0059	0.0813	0.1000	-0.0065 ± 0.0078	-0.0297	0
	CL	0.0986 ± 0.0035	0.0858	0.1000	-0.0018 ± 0.0026	-0.0105	0
	KB	0.0979 ± 0.0035	0.0879	0.1000	-0.0027 ± 0.0031	-0.0135	0
	PEI	0.0982 ± 0.0037	0.0853	0.1000	**-0.0014** ± 0.0020	-0.0082	0
	NPMS	0.0993 ± 0.0004	0.0986	0.0999	-0.0932 ± 0.0374	-0.1897	-0.0283
Levy (d=10)	EI	**0.0985** ± 0.0038	0.0827	0.1000	-0.2221 ± 0.1175	-0.4829	-0.0062
	GO-HI	0.0988 ± 0.0027	0.089	0.1000	-0.5405 ± 0.2004	-0.9911	-0.1866
	CL	0.0996 ± 0.0004	0.0978	0.1000	-0.0215 ± 0.0350	-0.1750	0
	KB	0.0987 ± 0.0043	0.0779	0.1000	**-0.0215** ± 0.0298	-0.1105	0
	PEI	0.0990 ± 0.0022	0.0885	0.1000	-3.2098 ± 0.2852	-3.9556	-2.5890
	NPMS	0.0996 ± 0.0003	0.0990	0.1000	-3.039 ± 0.8231	-5.0872	-1.8736

the case of EAVD, NPMS scheme has a clear tendency towards optimal values in initial EGO iteration. Compared to other parallel strategies, NPMS scheme reduces the cost of optimization by at least 62.3% and achieves 20.6% improvement in result accuracy. In the Boh case, NPMS scheme does not have good iterative convergence. Compared to EI strategy, NPMS scheme requires only 75.4% of I_s and 151.5% of N_s on average while improving *Res* by 65.2%. Compared to other parallel infilling strategies, NPMS scheme has the lowest computational cost and reduces at least 36.4%. In the Bar case, compared to other infilling strategies, the NPMS scheme reduces at least 22.9% of N_s and 14.6% of I_s, while the residual mean is shortened by 54.6%. CL strategy has better kriging model performance in comparison. As shown in Figure 14, at the first EGO iteration, NPMS generates nine candidate points, which are widely distributed. In comparison, the CL strategy is fixed to generate four candidate points. As the EGO iteration

TABLE 6. The optimization performance in group 1 and group 2.

Function	Strategy	Res			N_s			I_s		
		mean±std	min	max	mean±std	min	max	mean±std	min	max
Leo (d=2)	EI	0.0222 ± 0.0970	0	0.5352	28.3000 ± 7.9401	17	54	28.3000 ± 7.9401	17	54
	GO-HI	0.2146 ± 0.1577	0.0133	0.6777	**21.1667** ± 3.3742	17	32	21.1667 ± 3.3742	17	32
	CL	0.0161 ± 0.0111	0.0028	0.0583	44.6667 ± 23.9789	28	160	11.1667 ± 5.9947	7	40
	KB	0.1645 ± 0.0387	0.0389	0.1990	25.3333 ± 1.9179	24	28	**6.3333** ± 0.4795	6	7
	PEI	0.0164 ± 0.0125	0.0005	0.0427	47.8667 ± 3.0596	40	52	11.9667 ± 0.7649	10	13
	NPMS	**0.0119** ± 0.0196	0	0.0861	27.9667 ± 3.5862	17	35	13.2333 ± 2.1922	7	17
Alp (d=2)	EI	0.0044 ± 0.0046	0.0001	0.0147	69.5667 ± 2.4731	65	75	69.5667 ± 2.4731	65	75
	GO-HI	0.0095 ± 0.0112	0.0001	0.0412	64.5667 ± 3.1259	58	71	64.5667 ± 3.1259	58	71
	CL	0.0106 ± 0.0151	0	0.0553	60.8000 ± 18.0849	24	104	**15.2000** ± 4.5212	6	26
	KB	0.0013 ± 0.0024	0.0001	0.0100	105.2000 ± 5.9562	92	120	26.3000 ± 1.4890	23	30
	PEI	**0.0013** ± 0.0019	0.0001	0.0066	95.4667 ± 9.6123	72	112	23.8667 ± 2.4031	18	28
	NPMS	0.0038 ± 0.0058	0	0.0267	**51.6333** ± 3.8371	40	58	25.2667 ± 2.3479	20	30
Buk (d=2)	EI	6.1320 ± 3.4850	0.6490	12.5022	178.3333 ± 28.0681	144	266	178.3333 ± 28.0681	144	266
	GO-HI	8.0522 ± 4.0250	2.1141	16.4217	147.4000 ± 12.1786	122	175	147.4000 ± 12.1786	122	175
	CL	7.6126 ± 4.2485	0.6239	18.8112	171.2000 ± 31.7125	96	224	42.8000 ± 7.9281	24	56
	KB	7.7736 ± 3.3516	1.8068	13.6882	182.8000 ± 24.9502	120	232	45.7200 ± 6.2375	30	58
	PEI	7.7391 ± 4.1051	0.6358	15.2155	166.8000 ± 20.2645	136	208	**41.7200** ± 5.0661	34	52
	NPMS	**5.6399** ± 2.9097	1.3320	12.3313	**113.4333** ± 57.4244	40	279	54.1667 ± 28.4885	17	137
Cub (d=2)	EI	3.2207 ± 1.8966	0.3232	8.6466	282.9000 ± 5.9674	271	293	282.9000 ± 5.9674	271	293
	GO-HI	14.6299 ± 17.3183	0.1068	59.6449	**219.8000** ± 8.9420	198	236	219.8000 ± 8.9420	198	236
	CL	2.0856 ± 1.7512	0.0603	7.0230	454.4000 ± 31.5765	384	504	**113.6000** ± 7.8941	96	126
	KB	1.7449 ± 1.3773	0.0243	4.7014	538.1333 ± 13.6451	512	560	134.5333 ± 3.4113	128	140
	PEI	1.9463 ± 2.1115	0.0599	11.2466	522.9333 ± 19.0515	484	556	130.7333 ± 4.7629	121	139
	NPMS	**1.7142** ± 1.3996	0.1331	6.2850	280.4333 ± 13.2371	260	312	144.4333 ± 8.4596	129	168
Hol (d=2)	EI	0.0028 ± 0.0037	0	0.0195	229.2667 ± 70.5192	161	425	229.2667 ± 70.5192	161	425
	GO-HI	0.0581 ± 0.0794	0.0010	0.3313	149.4000 ± 61.1418	69	316	149.4000 ± 61.1418	69	316
	CL	0.0275 ± 0.0538	0	0.2806	166.8000 ± 69.0998	52	448	**41.7000** ± 17.2750	13	112
	KB	0.0022 ± 0.0039	0	0.0196	241.0667 ± 82.8122	144	524	60.2667 ± 20.7030	36	131
	PEI	**0.0021** ± 0.0027	0	0.0113	219.2000 ± 60.5210	152	424	54.8000 ± 15.1302	38	106
	NPMS	0.0162 ± 0.0132	0.0001	0.0583	**145.0000** ± 36.2282	64	214	83.9333 ± 24.0774	36	142
Cro (d=2)	EI	0.0009 ± 0.0005	0.0004	0.0025	147.5667 ± 31.7986	107	247	147.5667 ± 31.7986	107	247
	GO-HI	0.0049 ± 0.0051	0.0006	0.0214	138.9333 ± 24.2372	93	212	138.9333 ± 24.2372	93	212
	CL	0.0011 ± 0.0009	0.0004	0.0049	275.7333 ± 168.7371	52	616	68.9333 ± 42.1843	13	154
	KB	0.0009 ± 0.0005	0.0004	0.0023	243.4667 ± 79.3055	144	476	60.8667 ± 19.8264	36	119
	PEI	0.0014 ± 0.0017	0.0004	0.0072	288.8000 ± 105.3305	168	580	72.2000 ± 26.3326	42	145
	NPMS	**0.0005** ± 0.0003	0.0004	0.0016	**115.5333** ± 23.3160	78	194	**60.5333** ± 12.2326	40	97

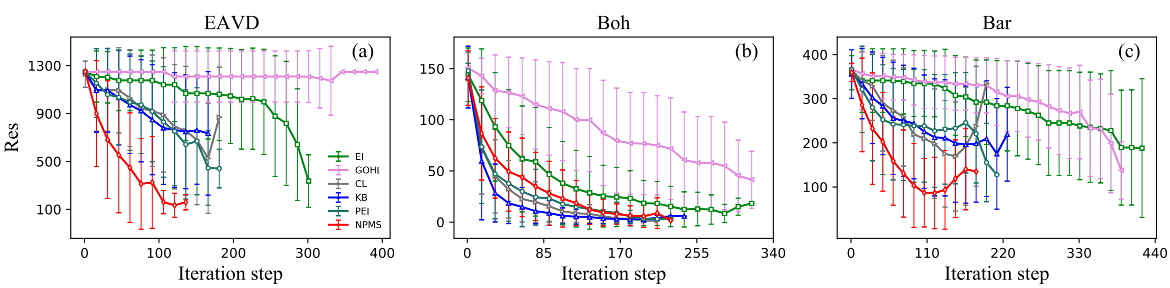


FIGURE 13. The residual convergence histories (mean + std) of different strategies in group 3. The horizontal solid line and dotted line represent the median and average values of each group of residuals, respectively, and the white dots represent outliers for each set of data.

TABLE 7. The optimization performance in group 3 and group 4.

Function	Strategy	Res			N_s			I_s		
		mean±std	min	max	mean±std	min	max	mean±std	min	max
EAVD (d=2)	EI	1025.8945 ± 410.9340	59.0236	1247.8480	239.1667 ± 41.7035	161	309	239.1667 ± 41.7035	161	309
	GO-HI	1208.2002 ± 217.1569	58.4299	1247.8480	339.7333 ± 28.8204	299	402	339.7333 ± 28.8204	299	402
	CL	808.4358 ± 523.9784	58.4627	1247.8480	607.0667 ± 87.4382	416	780	151.7667 ± 21.8596	104	195
	KB	656.7351 ± 485.1830	98.5620	1247.8480	590.2667 ± 66.2758	472	712	147.5667 ± 16.5689	118	178
	PEI	628.3088 ± 459.5476	43.1150	1247.8480	608.8000 ± 78.1539	444	780	152.2000 ± 19.5385	111	195
	NPMS	**168.7145** ± 116.5422	34.2246	509.8543	**222.4333** ± 55.3214	90	346	**95.2000** ± 23.2504	39	145
Boh (d=2)	EI	11.7189 ± 15.0011	0.6389	49.7938	**261.8667** ± 34.6597	189	330	261.8667 ± 34.6597	189	330
	GO-HI	51.2125 ± 39.0036	0.6870	150.7431	302.2667 ± 12.8276	274	325	302.2667 ± 12.8276	274	325
	CL	8.8722 ± 10.7008	0.4926	38.1390	624.1333 ± 103.0968	384	876	**156.0333** ± 25.7742	96	219
	KB	3.3884 ± 4.4218	0.0889	18.0787	707.3333 ± 116.0931	496	1012	176.8333 ± 29.0233	124	253
	PEI	6.6636 ± 7.9489	0.6099	38.2460	747.7333 ± 88.9785	552	952	186.9333 ± 22.2446	138	238
	NPMS	**2.7612** ± 3.1135	0.0970	14.8688	396.7333 ± 44.7213	285	469	197.4333 ± 21.6853	145	233
Bar (d=2)	EI	225.6782 ± 127.2151	29.9056	365.1407	375.4000 ± 31.2472	316	428	375.4000 ± 31.2472	316	428
	GO-HI	262.9216 ± 105.4551	71.7701	365.1407	358.6667 ± 27.9314	301	403	358.6667 ± 27.9314	301	403
	CL	172.7855 ± 119.4521	8.8329	365.1407	657.3333 ± 58.7158	528	796	164.3333 ± 14.6789	132	199
	KB	182.7480 ± 135.6031	10.0354	365.1407	797.2000 ± 69.9332	612	944	199.3000 ± 17.4833	153	236
	PEI	208.2317 ± 125.8893	8.6488	365.1407	700.9333 ± 86.7159	488	848	175.2333 ± 21.6790	122	212
	NPMS	**78.4190** ± 75.4158	0.6014	263.5427	**276.6667** ± 50.5141	161	375	**140.3000** ± 20.0931	98	182
Dix (d=4)	EI	4.5608 ± 3.5148	1.1825	16.0900	139.0000 ± 37.8062	102	226	139.0000 ± 37.8062	102	226
	GO-HI	5.8470 ± 6.9227	0.6111	36.2462	384.0333 ± 10.0567	364	404	384.0333 ± 10.0567	364	404
	CL	6.4936 ± 5.5885	1.0169	21.0338	95.4667 ± 14.0043	80	132	**23.8667** ± 3.5011	20	33
	KB	6.4338 ± 6.9963	0.8927	34.6958	100.5333 ± 16.4332	84	144	25.1333 ± 4.1083	21	36
	PEI	3.0189 ± 1.6736	0.8445	8.4242	521.2000 ± 50.4343	400	604	130.3000 ± 12.6086	100	151
	NPMS	**1.6280** ± 1.4432	0.3181	7.3564	**78.8333** ± 12.3765	65	111	71.8667 ± 11.7230	58	103
Hart (d=6)	EI	0.6306 ± 0.1864	0.2660	1.1551	269.4667 ± 23.5793	225	311	269.4667 ± 23.5793	225	311
	GO-HI	0.6188 ± 0.2158	0.1982	1.2379	336.8667 ± 40.0609	220	403	336.8667 ± 40.0609	220	403
	CL	0.5273 ± 0.1705	0.2279	0.8837	391.7333 ± 48.9327	320	524	97.9333 ± 12.2332	80	131
	KB	0.5629 ± 0.2336	0.1617	1.3502	291.8667 ± 72.9023	40	376	**72.9667** ± 18.2256	10	94
	PEI	0.4852 ± 0.1626	0.2045	0.7493	359.4667 ± 41.5391	244	452	89.8667 ± 10.3848	61	113
	NPMS	**0.3728** ± 0.1496	0.1147	0.6803	**269.3667** ± 36.7344	179	354	141.4667 ± 31.2859	59	225
Trid (d=8)	EI	1784.2017 ± 734.1346	524.6351	2981.7321	**180.7000** ± 28.6238	111	234	180.7000 ± 28.6238	111	234
	GO-HI	1943.6241 ± 539.7416	811.5371	2960.2867	219.5333 ± 31.2043	141	289	219.5333 ± 31.2043	141	289
	CL	1427.8249 ± 435.6131	602.7944	2426.4359	663.4667 ± 116.7549	448	840	165.8667 ± 29.1887	112	210
	KB	1357.0238 ± 383.5783	444.4648	2183.4864	668.5333 ± 117.3441	360	848	167.1333 ± 29.336	90	212
	PEI	1459.9727 ± 474.1365	528.2705	2118.6522	662.6667 ± 127.7512	328	920	165.6667 ± 31.9378	82	230
	NPMS	**1297.0663** ± 409.4498	576.6048	2160.7227	201.6000 ± 19.4823	146	221	**91.4333** ± 19.3349	37	113
Levy (d=10)	EI	22.4600 ± 5.2451	8.9897	31.8627	239.6667 ± 44.0230	165	339	239.6667 ± 44.0230	165	339
	GO-HI	24.6122 ± 5.4479	13.1637	33.6316	224.4667 ± 37.0542	138	294	224.4667 ± 37.0542	138	294
	CL	15.2605 ± 4.5493	7.7262	22.8390	739.2000 ± 126.2440	500	1108	184.8000 ± 31.5610	125	277
	KB	15.1867 ± 4.0880	5.0166	22.8545	733.8667 ± 86.4039	560	868	183.4667 ± 21.6010	140	217
	PEI	23.3057 ± 7.2687	8.7548	39.0612	722.0000 ± 92.3905	524	888	180.5000 ± 23.0976	131	222
	NPMS	**9.1508** ± 2.9711	4.0742	17.4957	**204.4333** ± 23.4222	156	244	**149.3333** ± 16.957	113	179

increases, the number of candidate points generated by NPMS decreases and is uniformly distributed, while CL appears as aggregating points. And the cumulative total number of candidate points in the CL strategy is significantly more than NPMS scheme. By comparison, because of its adaptive and global nature, NPMS makes EGO optimization more effective. NPMS scheme can quickly establish a global search at the beginning of optimization, generating more candidate points in uncertain regions and increasing the likelihood of finding the optimal value. During EI function convergence, NPMS scheme dynamically changes the number of points selected, leading to fewer candidate points than CL strategy after 80^{th} iteration of EGO. In addition, NPMS scheme maintains a balance between the amount of information and diversity of candidate points that each point kept away from each other, which is less likely to cause points aggregation. Above, NPMS scheme has the best result ranking in large input space cases.

For group 4, the situation is similar to group 3, where NPMS outperforms the other strategies in general, and the convergence history of strategy residuals shown in Figure 15. In Dix and Hart's case, CL and KB reach EGO stopping conditions faster, while NPMS scheme has better optimization results and costs. In the Dix case, the convergence of

TABLE 8. The Friedman test results of statistical difference between strategies.

Benchmark	Result	Strategy rank					
		EI	GO-HI	CL	KB	PEI	NPMS
group1	Res	3.33 (2)	5.67 (6)	3.67 (4)	3.83 (5)	2.83 (2)	1.67 (1)
	N_s	15.83 (4)	12.50 (2)	14.67 (3)	16.33 (5)	16.33 (5)	12.00 (1)
	I_s	15.83 (6)	12.50 (5)	7.67 (1)	8.67 (3)	8.00 (2)	9.67 (4)
group2	Res	3.50 (4)	6.00 (6)	4.33 (5)	2.17 (2)	3.00 (3)	2.00 (1)
	N_s	15.17 (3)	12.17 (2)	15.67 (4)	17.33 (6)	16.67 (5)	11.67 (1)
	I_s	15.17 (6)	12.17 (5)	7.67 (1)	8.67 (2)	8.67 (2)	9.00 (4)
group3	Res	10.33 (5)	11.33 (6)	8.00 (3)	7.67 (2)	8.33 (4)	2.33 (1)
	N_s	11.17 (2)	11.83 (3)	14.67 (4)	15.33 (5)	16.00 (6)	10.67 (1)
	I_s	11.17 (5)	11.83 (6)	4.33 (1)	5.67 (3)	6.00 (4)	4.33 (1)
group4	Res	4.500 (5)	5.250 (6)	3.705 (4)	3.250 (2)	3.250 (2)	1.000 (1)
	N_s	2.500 (2)	3.500 (3)	4.750 (5)	4.250 (4)	4.750 (5)	1.250 (1)
	I_s	5.250 (5)	5.750 (6)	2.750 (4)	2.500 (3)	2.500 (2)	2.250 (1)
Average	Res	7.50 (5)	8.25 (6)	6.75 (4)	6.25 (2)	6.25 (2)	4.00 (1)
	N_s	11.75 (2)	13.25 (3)	14.50 (5)	13.75 (4)	15.25 (6)	9.75 (1)
	I_s	11.75 (5)	13.25 (6)	7.25 (3)	7.00 (2)	7.75 (4)	6.75 (1)

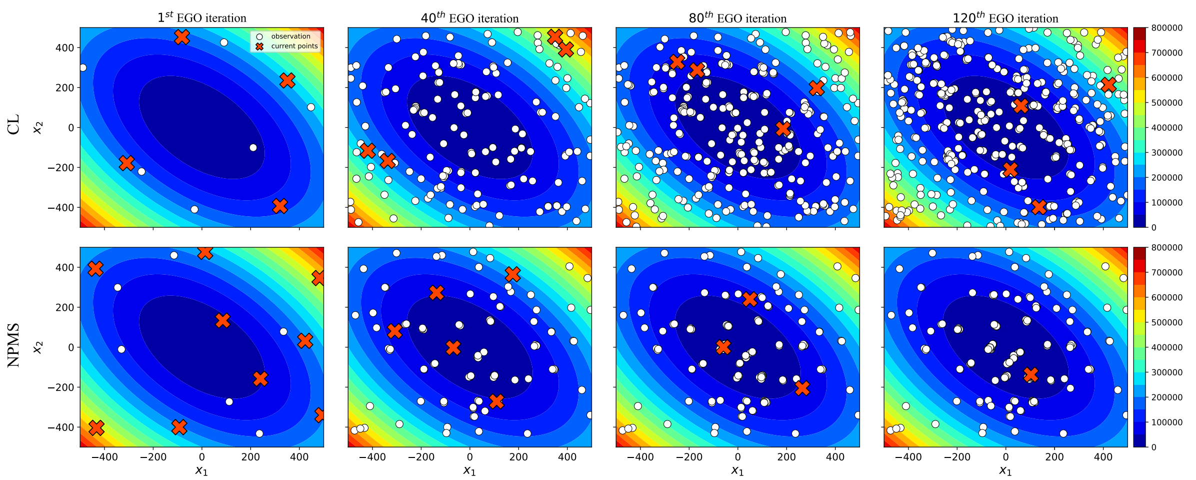


FIGURE 14. The candidate points generated by CL strategy and NPMS scheme in the Bar case, where red points are the candidate points generated by current EGO iteration, and white points are accumulated candidate points.

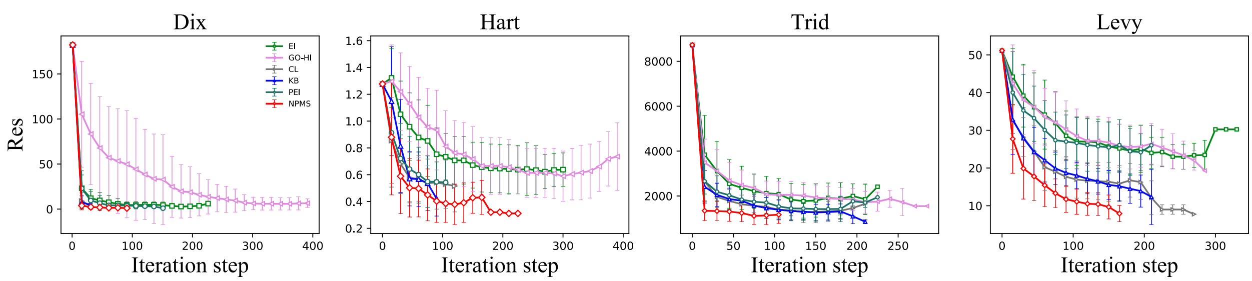


FIGURE 15. The residual convergence histories (mean + std) of infilling strategies in group 4.

NPMS scheme residuals has a pronounced downward trend. Compared to single-point infilling strategies, NPMS scheme needs only 56.7% of N_s and 51.7% of I_s. Compared to CL strategy, NPMS scheme has slower optimization iterations and saves 17.4% of computational cost. In the Hart case, NPMS scheme has the worst convergence compared to other parallel infilling strategies but achieves smaller residual values and saves at least 7.71% of computational cost. In the

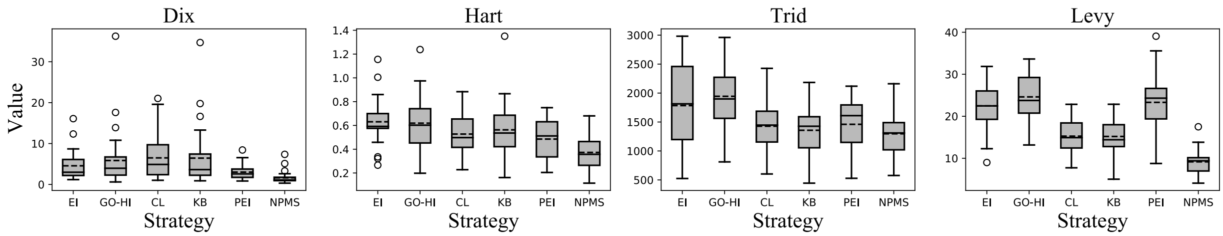


FIGURE 16. **The boxplots of the residuals for the different strategies in group 4.**

Trid case, all infilling strategies fail to find good optimal values, and NPMS scheme has well average residual value. Compared to EI strategy, NPMS scheme can find good results in the initial optimization iterations, and needs 112% of N_s and 55.6% of I_s. In the Levy case, NPMS scheme is more likely to find better values, needs 8.92% less computational cost on average, and 17.3% fewer iterations than other parallel infilling strategies. On average, NPMS scheme has the best ranking, saving on average 45.8% of optimization cost and 25.7% of iterations while improving optimization results by 17.7%.

Overall, Friedman test shows that NPMS scheme is the best at finding minima and cost savings. In small and medium input space size cases, NPMS has a slight advantage in optimization efficiency and performs poorly in constructing kriging models. On the other hand, when dimensionality or input size rise, EI function focuses more on exploring uncertainty regions. NPMS scheme samples from global interest areas with much EI function calculation and reduced optimization evaluation make the scheme save costs and find good results.

V. CONCLUSION

We have proposed an extended adaptive parallel infilling strategy of Expected Improvement (EI) with NPMS scheme consisting of two processing stages, that is, PMC sampling and DBSCAN clustering. The scheme can adaptively generate samples in sub-domains with high EI values and finally obtain the desired multi-point in a convergent manner. When benchmarked against the three-hump hump function with high drop characteristics, our scheme improves the result accuracy by 14.6% and reduces the cost of optimization evaluation by 15.8%, relative to conventional EI strategy. In further tests with other five strategies, NPMS with optimized parameters can achieve an increase in result accuracy and a reduction in the number of candidate points across 13 functions with different input spaces, difficulties, and dimensions. When facing poor kriging prediction, especially in large input spaces and high-dimensional functions, NPMS can surprisingly give an average of 72% higher optimization result accuracy than the other five strategies. Taking advantage of broad sampling and adaptive clustering, in fact, NPMS can ensure that candidate points come from promising sub-domains and dynamically control the number of candidate points, making the extended EI strategy more efficient and less costly. Therefore, our model provides a novel idea for the exploration-exploitation balance in EI strategy, which establishes the basis for solving more complex EGO problems in the future.

REFERENCES

[1] L. Aydin, O. Aydin, H. S. Artem, and A. Mert, "Design of dimensionally stable composites using efficient global optimization method," *Proc. Inst. Mech. Eng., L, J. Mater., Des. Appl.*, vol. 233, no. 2, pp. 156–168, Feb. 2019.

[2] C. Weihs, S. Herbrandt, N. Bauer, K. Friedrichs, and D. Horn, "Efficient global optimization: Motivation, variations, and applications," *Arch. Data Sci.*, vol. 2, no. 1, p. 26, 2017, doi: 10.5445/KSP/1000058749/01.

[3] Y. Xiao, J. Ning, Z. Xiong, and H. Qin, "Batch sequential adaptive designs for global optimization," *J. Korean Stat. Soc.*, vol. 51, no. 3, pp. 1–23, 2022.

[4] J. Wilson, V. Borovitskiy, A. Terenin, P. Mostowsky, and M. Deisenroth, "Efficiently sampling functions from Gaussian process posteriors," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 10292–10302.

[5] J. Mockus, V. Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," *Towards Global Optim.*, vol. 2, nos. 117–129, p. 2, 1978.

[6] S. Khalilpourazari, S. Khalilpourazary, A. Ö. Çiftçioğlu, and G.-W. Weber, "Designing energy-efficient high-precision multi-pass turning processes via robust optimization and artificial intelligence," *J. Intell. Manuf.*, vol. 32, no. 6, pp. 1621–1647, Aug. 2021.

[7] S. Khalilpourazari and H. Hashemi Doulabi, "Robust modelling and prediction of the COVID-19 pandemic in Canada," *Int. J. Prod. Res.*, vol. 2022, pp. 1–17, Jun. 2021.

[8] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, Dec. 1998.

[9] S. Jeong and S. Obayashi, "Efficient global optimization (EGO) for multi-objective problem and data mining," in *Proc. IEEE Congr. Evol. Comput.*, vol. 3, Sep. 2005, pp. 2138–2145.

[10] N. Henkenjohann and J. Kunert, "An efficient sequential optimization approach based on the multivariate expected improvement criterion," *Quality Eng.*, vol. 19, no. 4, pp. 267–280, Oct. 2007.

[11] R. G. Regis, "Trust regions in Kriging-based optimization with expected improvement," *Eng. Optim.*, vol. 48, no. 6, pp. 1037–1059, Jun. 2016.

[12] K. Zhang, Z. Wang, G. Chen, L. Zhang, Y. Yang, C. Yao, J. Wang, and J. Yao, "Training effective deep reinforcement learning agents for real-time life-cycle production optimization," *J. Petroleum Sci. Eng.*, vol. 208, Jan. 2022, Art. no. 109766.

[13] Y. Han, X. Xu, Y. Zhao, X. Wang, Z. Chen, and J. Liu, "Impact of consumer preference on the decision-making of prefabricated building developers," *J. Civil Eng. Manage.*, vol. 28, no. 3, pp. 166–176, Feb. 2022.

[14] J. Zhang, Y. Tang, H. Wang, and K. Xu, "ASRO-DIO: Active subspace random optimization based depth inertial odometry," *IEEE Trans. Robot.*, early access, Oct. 11, 2022, doi: 10.1109/TRO.2022.3208503.

[15] H. J. Kushner, "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise," *J. Basic Eng.*, vol. 86, no. 1, pp. 97–106, Mar. 1964.

[16] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *J. Global Optim.*, vol. 21, no. 4, pp. 345–383, 2001.

[17] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," 2009, *arXiv:0912.3995*.

[18] J. W. van Groenigen, "The influence of variogram parameters on optimal sampling schemes for mapping by Kriging," *Geoderma*, vol. 97, nos. 3–4, pp. 223–236, Sep. 2000.

[19] S. S. Garud, I. A. Karimi, and M. Kraft, "Design of computer experiments: A review," *Comput. Chem. Eng.*, vol. 106, pp. 71–95, Nov. 2017.

[20] C. Qin, D. Klabjan, and D. Russo, "Improving the expected improvement algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–15.

[21] Z. Guo, Y.-S. Ong, and H. Liu, "Calibrated and recalibrated expected improvements for Bayesian optimization," *Struct. Multidisciplinary Optim.*, vol. 64, no. 6, pp. 3549–3567, Dec. 2021.

[22] W. Chen, K. Chiu, and M. Fuge, "Aerodynamic design optimization and shape exploration using generative adversarial networks," in *Proc. AIAA Scitech Forum*, Jan. 2019, p. 2351.

[23] M. Schonlau, W. J. Welch, and D. R. Jones, "Global versus local search in constrained optimization of computer models," *Lect. Notes-Monograph Ser.*, vol. 34, pp. 11–25, Jan. 1998.

[24] A. Sóbester, S. J. Leary, and A. J. Keane, "On the design of optimization strategies based on global response surface approximation models," *J. Global Optim.*, vol. 33, no. 1, pp. 31–59, Sep. 2005.

[25] L. Lv, M. Shi, X. Song, W. Sun, and J. Zhang, "A fast-converging ensemble infilling approach balancing global exploration and local exploitation: The go-inspired hybrid infilling strategy," *J. Mech. Design*, vol. 142, no. 2, pp. 1–13, Feb. 2020.

[26] M. Schonlau, *Computer Experiments and Global Optimization*. UWSpace, 1997. [Online]. Available: http://hdl.handle.net/10012/190

[27] D. Zhan and H. Xing, "Expected improvement for expensive optimization: A review," *J. Global Optim.*, vol. 78, no. 3, pp. 507–544, Nov. 2020.

[28] Y. Zhang, S. Wang, C. Zhou, L. Lv, and X. Song, "A fast active learning method in design of experiments: Multipeak parallel adaptive infilling strategy based on expected improvement," *Struct. Multidisciplinary Optim.*, vol. 64, no. 3, pp. 1259–1284, Sep. 2021.

[29] H. Liu, Y.-S. Ong, and J. Cai, "A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design," *Struct. Multidisciplinary Optim.*, vol. 57, no. 1, pp. 393–416, 2018.

[30] Y. Mao, Y. Zhu, Z. Tang, and Z. Chen, "A novel airspace planning algorithm for cooperative target localization," *Electronics*, vol. 11, no. 18, p. 2950, Sep. 2022.

[31] A. Sóbester, S. J. Leary, and A. J. Keane, "A parallel updating scheme for approximating and optimizing high fidelity computer simulations," *Struct. Multidisciplinary Optim.*, vol. 27, no. 5, pp. 371–383, Jul. 2004.

[32] D. Ginsbourger, R. Le Riche, and L. Carraro, "A multi-points criterion for deterministic parallel global optimization based on Gaussian processes," Version v1, Tech. Rep. hal-00260579, Mar. 2008. [Online]. Available: https://hal.science/hal-00260579 and https://hal.science/hal-00260579/file/qEI_Ginsbourger_leriche_carraro_0303_2008.pdf

[33] D. Ginsbourger, R. L. Riche, and L. Carraro, "Kriging is well-suited to parallelize optimization," in *Computational Intelligence in Expensive Optimization Problems*. Cham, Switzerland: Springer, 2010, pp. 131–162.

[34] D. Zhan, J. Qian, and Y. Cheng, "Pseudo expected improvement criterion for parallel EGO algorithm," *J. Global Optim.*, vol. 68, no. 3, pp. 641–662, Jul. 2017.

[35] Y.-M. Xie, Y.-H. Guo, F. Zhang, Y.-P. Yue, M.-Q. Feng, and J.-B. Zhao, "An efficient parallel infilling strategy and its application in sheet metal forming," *Int. J. Precis. Eng. Manuf.*, vol. 21, no. 8, pp. 1479–1490, Aug. 2020.

[36] M. Gobert, J. Gmys, N. Melab, and D. Tuyttens, "Adaptive space partitioning for parallel Bayesian optimization," in *Proc. 18th Int. Conf. High Perform. Comput. Simulation*, 2021, pp. 1–8.

[37] D. Zhan, Y. Meng, and H. Xing, "A fast multipoint expected improvement for parallel expensive optimization," *IEEE Trans. Evol. Comput.*, vol. 27, no. 1, pp. 170–184, Feb. 2023.

[38] M. A. Beaumont, J.-M. Cornuet, J.-M. Marin, and C. P. Robert, "Adaptive approximate Bayesian computation," *Biometrika*, vol. 96, no. 4, pp. 983–990, Dec. 2009.

[39] E. Schubert, J. Sander, M. Ester, H. Kriegel, and X. Xu, "DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN," *ACM Trans. Database Syst.*, vol. 42, no. 3, pp. 1–21, 2017.

[40] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.*, vol. 7, p. e623, Jul. 2021.

[41] T. Chouard, "The go files: AI computer wins first match against master go player," *Nature*, Mar. 2016, doi: 10.1038/nature.2016.19544.

[42] M. Jamil and X.-S. Yang, "A literature survey of benchmark functions for global optimisation problems," *Int. J. Math. Modeling Numer. Optim.*, vol. 4, no. 2, pp. 150–194, 2013.

[43] A. B. Owen, "Orthogonal arrays for computer experiments, integration and visualization," *Statistica Sinica*, vol. 2, no. 2, pp. 439–452, 1996. Accessed: Feb. 16, 2023. [Online]. Available: http://www.jstor.org/stable/24304869

[44] Z. Wen, H. Pei, H. Liu, and Z. Yue, "A sequential Kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability," *Rel. Eng. Syst. Saf.*, vol. 153, pp. 170–179, Sep. 2016.

[45] Z. Wang, H. Xie, Q. Luo, Q. Li, and G. Sun, "Optimizaition for formability of plain woven carbon fiber fabrics," *Int. J. Mech. Sci.*, vol. 197, May 2021, Art. no. 106318.

[46] Y. Li, J. Shi, H. Cen, J. Shen, and Y. Chao, "A Kriging-based adaptive global optimization method with generalized expected improvement and its application in numerical simulation and crop evapotranspiration," *Agricult. Water Manage.*, vol. 245, Feb. 2021, Art. no. 106623.

[47] J. L. Loeppky, J. Sacks, and W. J. Welch, "Choosing the sample size of a computer experiment: A practical guide," *Technometrics*, vol. 51, no. 4, pp. 366–376, Nov. 2009.

[48] J. A. Colton and K. M. Bower, "Some misconceptions about R2," *Int. Soc. Six Sigma Professionals*, vol. 3, no. 2, pp. 20–22, 2002.

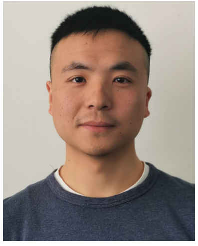
[49] E. Hlawka, "Funktionen von beschränkter variatiou in der theorie der gleichverteilung," *Annali Matematica Pura Applicata*, vol. 54, no. 1, pp. 325–333, 1961.

[50] A. Verma, K. Wong, and A. L. Marsden, "A concurrent implementation of the surrogate management framework with application to cardiovascular shape optimization," *Optim. Eng.*, vol. 21, no. 4, pp. 1487–1536, Dec. 2020.

[51] J. Wang, S. C. Clark, E. Liu, and P. I. Frazier, "Parallel Bayesian global optimization of expensive functions," *Oper. Res.*, vol. 68, no. 6, pp. 1850–1865, Nov. 2020.

YU HU was born in Zhejiang, China. He received the B.S. degree in computer science and engineering from Shaoxing University, in 2020. He is currently pursuing the M.S. degree in computer technology from Ningbo University.

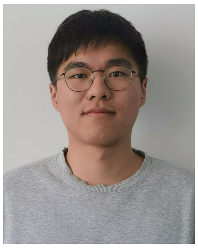
His research interests include Bayesian inference using approximate sampling, global optimization, and data science.



YAOLIN GUO received the Ph.D. degree from the School of Materials Science and Engineering, Northwestern Polytechnical University, China, in 2015. From January 2016 to October 2019, he worked as a Postdoctoral Fellow at the Ningbo Institute of Materials Science, Chinese Academy of Sciences. During his work at the Advanced Energy Materials Engineering Laboratory, he mainly undertook the design and development of multi-scale calculation software for materials, and participated in and completed two national key research and development programs. He is currently an Engineer with the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences. He is currently implementing the optimal design of materials using computer technology through the strategies of the materials genome initiative. He has published over 30 articles in journals, such as *Acta Materialia*, *Scripta Materialia*, *Computer Physics Communications*, *Soft Matter*, *Philosophical Magazine*, and *Computational Materials Science*.

ZHEN LIU received the master's degree in materials science and engineering in Grenoble National Institute of Technology, France, in 2014, the doctor degree in material physics and chemistry from Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, in 2019. From 2008 to 2012, he studied nuclear engineering and technology at North China Electric Power University. He is currently an Associate Researcher. At present, he is mainly engaged in the application of molecular dynamics methods and artificial intelligence algorithms in the field of multi-scale coupling simulation of material properties. He has published more than ten papers in *Scripta Materialia*, *Journal of Alloys and Compounds*, and *Ceramics International*.

YIFAN LI is currently pursuing degree with College of Materials Science and Chemical Engineering, Harbin Engineering University. He is mainly engaged in theoretical research in the field of nuclear energy materials, using first principles, molecular dynamics, phase fields, and rate theory to study the microstructure, interfaces, and service properties of nuclear energy materials from different aspects and levels.



ZHEYU HU was born in Zhejiang, China. He received the B.S. degree from Quzhou University, in 2020. He is currently pursuing the M.S. degree in computer technology from Ningbo University.

His research interests include artificial intelligence and data science.

DIWEI SHI received the doctor's degree from ShanghaiTech University, in 2021. From 2009 to 2013, he studied materials science and engineering at Xi'an Jiaotong University. He is currently an Assistant Professor. He is mainly engaged in the prediction of material structures and properties, and more than ten papers have been published in *Computational Materials Science*, *Physical Chemistry Chemical Physics*, and the *Journal of Physical Chemistry*.

MORAN BU received the B.Eng. degree in engineering mechanics from the Harbin Institute of Technology, Harbin, China, and the M.E. degree in materials engineering from the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. He is currently pursuing the Ph.D. degree in ecology from the Northeast Institute of Geography and Agroecology, Chinese Academy of Sciences, Changchun, China. His research interests include computational materials science and bioinformatics.

SHIYU DU received the Ph.D. degree from the Department of Chemistry, Purdue University, USA, in 2009. From July 2009 to December 2013, he served as a postdoctoral researcher and a visiting scientist at Los Alamos National Laboratory, USA. During his stay in Los Alamos National Laboratory, he mainly undertook the theoretical calculation of key physical properties of nuclear fuel and nuclear material. He returned to China working full-time, in January 2014, and established a research team of ''Theoretical design and performance simulation of energy materials'' at the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, where he is a Doctoral Supervisor and currently a Professor. He is the Chief Scientist of the National Key Research and Development Program. He is currently working in the structural design and characterization of various energy and structural materials such as nuclear materials by the strategy of materials genome initiative and the multiscale computer simulation method. He has published more than 200 peer-reviewed research articles in *Nature Communications*, *Proceedings of the National Academy of Sciences of the United States of America*, *Journal of the American Chemical Society*, *Angewandte Chemie International Edition*, *ACS Nano*, *Nanoscale*, *The Journal of Physical Chemistry Letters*, and other international scientific research journals.