Journal of Network Intelligence ©2025 ISSN 2414-8105
Taiwan Ubiquitous Information Volume **10**, Number **4**, November **2025**

Exploration and Application of Probabilistic Trust Domain Methods in Bayesian Optimization

Li-Juan Zhao*

Nanjing Vocational Institute of Railway Technology
Nanjing 210031, P. R. China
zzlljj210@163.com

Hai Liu

Yana Royal Polytechnic University
Chiang Mai 50000, Thailand
wn0026@163.com

*Corresponding author: Li-Juan Zhao
Received September 5, 2024, revised April 12, 2025, accepted July 22, 2025.

Abstract. *Bayesian optimization (BO) needs a large number of sample points to find the ideal solution in the case of high-dimensional objective function, and it is tough to achieve good optimization outcome in practical applications. To deal with the above issues, this paper applies and explores how to improve BO algorithm with Probabilistic Trust Region (PTR). The PTR algorithm is designed to address the issue that the traditional Trust Region (TR) algorithm takes a long time to solve the subproblem. The probabilistic interpolation strategy is adopted to select the overlapping points with good properties of interpolation points set, and the initial augmenting Lagrange multiplier is modified to reduce the iterative time of TR. Then, two PTRs with dynamic size change are used to control the approach to focus more on partial search without losing the ability of global optimization. One PTR is centered on the current optimal solution and trains the Gaussian process inside the sphere. The new evaluation points selected by the other PTR during each iteration are located within the hyperrectangle. Nextly, the method of random search is adopted to optimize the collection function, which avoids the cost of optimizing the collection function. The experiment indicates that the gap measurement degree of the designed algorithm PTRBO is 0.84, which is higher than the comparison model. The experimental outcome is consistent with theoretical analysis, implying that PTRBO has good global optimization ability and robustness.*
Keywords: Bayesian optimization; Probabilistic interpolation; Trust region; Random search; Acquisition function

1. **Introduction.** Stochastic optimization problems are ubiquitous in actual production and engineering practice, such as logistics and supply chain [1], manufacturing [2], etc. However, in complex system design and engineering practice, the establishment of accurate and reliable simulation models usually requires a large amount of test data support, and complex test processes require expensive costs. For such complex design problems with black box and high evaluation cost, Bayesian Optimization (BO) method can efficiently search the design space and find the optimal or near-optimal solution on the basis of a small amount of sample data through the modeling method based on Bayesian inference. Effectively resolve the trade-off between high test costs and accurate simulation models. With the research and growth in recent years, BO algorithm has become one of the

mainstream methods in the fields of stochastic simulation [3], machine learning [4] and reinforcement learning [5].

1.1. **Related work.** Haskell et al. [6] first proposed the framework of modern BO and used Wiener process to solve the optimization problem under one-dimensional constraints. Mockus [7] proposed kriging, an algorithm that uses linear model modeling to solve high-dimensional optimization problems, and adopts Gaussian process to model the problems. At the same time, a BO algorithm using Gaussian process for problem modeling has been widely applied to gradient-free information optimization and experimental design and other related problems [8], and has achieved brilliant results in practical application and theoretical innovation. Moriconi et al. [9] used the Gaussian confidence upper bound information as the acquisition function for the BO algorithm of Gaussian process modeling. In theory, the number of iterations is sublinearly related when solving the algorithm. Obayya et al. [10] proposed the hyperparameter adjustment of CNN and other machine learning algorithms by using the BO algorithm of Gaussian process modeling.

Furthermore, BO is mainly used to solve the problem of low-dimensional optimization, but it is challenging to extend BO to high dimensions and a large number of sampling points. With the increase of the target operation domain's dimension, the size of the search space increases exponentially, resulting in a dimensional disaster. Luong et al. [11] decomposed the original high-dimensional target operation into numerous low-dimensional operations, then optimized the collection function with an efficient information transfer algorithm. Wang et al. [12] adopted a Fourier feature approximation approach to approximate the exponential square covariance function, and the approximate error would decrease exponentially with the increase of the amount of features. In addition, the work of extending BO to higher dimensions also includes [13, 14, 15].

In these methods, the high-dimensional objective function can be decomposed into a series of low-dimensional subfunctions. However, due to the existence of a large number of Gaussian processes, it is not feasible to sample a large number of algorithms. In addition, it is often difficult to optimize non-convex acquisition functions when using BO for high-dimensional problems. To cope with the above issues, gradient method [16], quasi-newton method [17] and Trust Region (TR) method [18] are the most important optimization methods. For the optimization of unconstrained optimization problems, TR method opens up a new idea for BO because of its good convergence and robustness. Liu [19] adopted TR-based Gaussian process as a probabilistic proxy model. Zhou [20] suggested a TR based BO algorithm, which simultaneously solved a large number of observation points, high dimensionality of input variables of the objective function, and batch generation of sample points with balanced diversity and accuracy. In addition, Diouane [21] offered TREGO, a local BO algorithm based on TR, which showed better performance when BO was used to process high-dimension and a large number of sampling points.

1.2. **Contribution.** Through the analysis of the research status, it can be seen that there is no general agent model in the existing BO that can be applied to different optimization problems, and it is difficult to deal with high-dimensional black box optimization problems, resulting in low global optimization ability. Therefore, this paper designs an improved BO algorithm based on TR algorithm with good convergence. Firstly, the Probabilistic Trust Region (PTR) algorithm is designed. By establishing the constraint violation function, the probabilistic interpolation strategy is used to select the overlapping points with good properties of the interpolation points set. By analyzing the relationship between the update of the model between the iteration of TR algorithm and the initial multiplier of the next iteration, the iteration times and iteration time of TR are reduced.

Then, two dynamic PTRs are used to control the algorithm, and one TR is used to limit the amount of instances used to train the Gaussian pass in the iterative process. Another PTR is adopted to limit the amount of the search space to generate new sample points. Finally, the random search method is used to optimize the collection function, and the sampling point is extended to batch sampling, which reduces the algorithm overhead. Theoretical analysis and experimental outcome indicate that the designed PTRBO algorithm has stronger local search and global optimization capabilities.

2. Theoretical analysis.

2.1. **Bayesian optimization.** Different from some traditional model methods, BO is based on probabilistic modeling and is used to solve complex optimization problems with black box and high evaluation cost [22]. The black box function diagram is shown in Figure 1. When the dimensionality of the objective function is not high, BO can find the ideal solution with only a few iterations, but for optimization problems with high dimensionality, BO performs poorly. This paper will use the optimized trust region algorithm to improve its optimization performance. The BO method includes the proxy model of the objective function and the collection function.

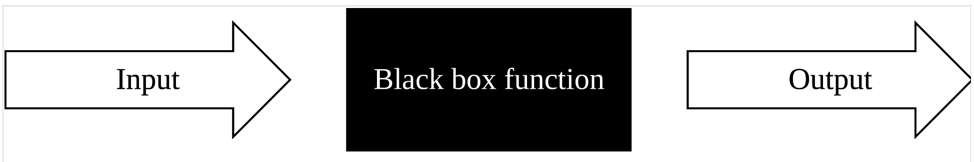

Figure 1. Diagram of black box function

(1) The proxy model can be categorized into parametric model and non-parametric model in terms of the amount of model parameters. The normally used parametric models contain Betta-Bernoulli model and linear model, etc., while the non-parametric models include Gaussian process and random forest, etc. Among them, the Gaussian process based on probability theory has universality and efficiency compared with other models, and is widely used in BO [23].

(2) The collection function is a function used to effectively explore the state space in the information of the agent model. In BO, the selection of the obtaining function is usually relied on the uncertainty of the agent model, the approximation accuracy, and the nature of the objective function.

2.2. **Trust region algorithm.** TR algorithm in the optimization algorithm process to find the displacement of each iteration, and then determine the new iteration point, and the traditional linear search method is different from the ordinary linear search is to first generate the search direction, and then determine the search step size, and TR is directly to determine the displacement to produce a new iteration point. Therefore, TR is a good local optimization method, especially for some optimization problems with very stable performance [24].

Generally, TR is a polycellular body focused on the existing optimal search, and its scale is constantly adapted with the epochs of the approach, as implied in Figure 2. The essence of TR is to improve a relatively easy partial approximation of the original target operation, and the initial issue is transformed into an improved issue with constraints.

$$\min_{d} \; m_k(d) \; = \; f_k + g_k^T d + \tfrac{1}{2} d^T B_k d \quad \text{s.t.} \;\; \|d\| \le \Delta_k \tag{1}$$

where f_k, g_k^T and B_k respectively represent the value of the objective function $f(x)$ at the current iteration point x_k, the gradient and the approximation matrix of the Hesse matrix, and Δ_k represents the radius of the TR.

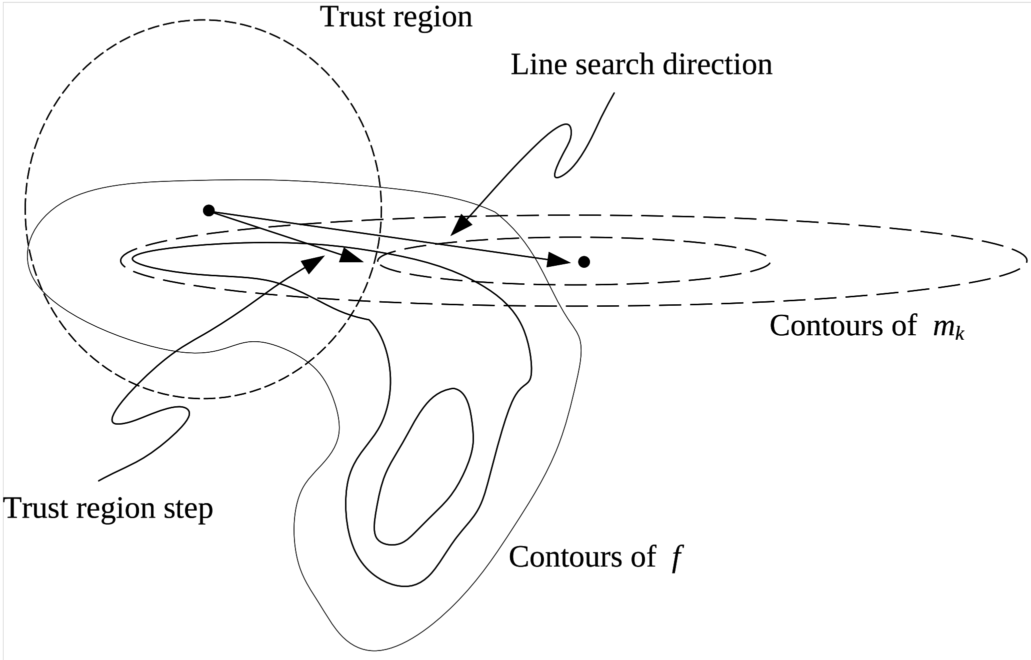

Figure 2. Schematic diagram of TR algorithm

Assuming that s_k is the solution to the above issue, the values of the evaluation function r_k below are used to determine the values of the next iteration point x_{k+1} and the new TR radius Δ_{k+1}.

$$r_k = \frac{f(x_k) - f(x_k + s_k)}{m_k(d) - m_k(s_k)} \tag{2}$$

The larger the value of r_k, the better the current approximation of the objective function. Through r_k, s_k and Δ_k in the iteration process, and the hyperparameter λ_i is required in the adjustment algorithm, the selection strategy for the next iteration point is shown as follows.

$$x_{k+1} = \begin{cases} x_k + s_k, & r_k \leq \lambda_i \\ x_k, & r_k \geq \lambda_i \end{cases} \tag{3}$$

3. **Trust region algorithm optimization based on probabilistic interpolation strategy.**

3.1. **Selection of iteration points by trust region algorithms.** The traditional TR algorithm first constructs the approximate model of the original problem from the given initial point, and then iteratively solves the sub-problem in TR to find the global minimum point of the original problem. This further increases the amount of computation. To address the above issues, this paper suggests probabilistic TR (PTR), and uses probabilistic interpolation (PT) strategy [25] to screen the set of interpolation points and find the iteration points with good properties to speed up the decline of function values. In order to reduce the time of solving the subproblem, the deficiency of the initial value returned by the Lagrange multiplier is corrected.

In the iterative process of traditional TR algorithm, many intermediate iteration points are interpolating points, which lacks full use of the information of the existing interpolating points. To solve this issue, this paper establishes the constraint violation function, uses PT strategy to construct the constraint violation function l and the constraint function obstacle critical value $k_{\max}$, so that the less feasible point meeting certain conditions is searched as the intermediate iteration point, and gradually reduces the constraint obstacle critical value, with the aim of the algorithm can quickly find the feasible solution to the problem. The constraint violation function $l : \mathbb{R}^n \to \mathbb{R}_+ \cup \{+\infty\}$ is the aggregate of all the constraint violation functions. For example, when the objective function constraint is $c_i(x) \geq 0$ $(i = 1, 2, \ldots, m)$ and the interval constraint is $x \in X$, the constraint violation function is as follows.

$$l(x) = \begin{cases} \sum_{i=1}^{m} \big(\max\{-c_i(x),\, 0\}\big)^2, & x \in X, \\ +\infty, & \text{otherwise.} \end{cases} \tag{4}$$

The PT strategy relies on the constraint barrier critical value l_k to filter out the points of $l(x) > l_k$ first, and then find the iteration points with better properties.

$$x_k = \arg\min_{x \in X}\ f(x) \tag{5}$$

where $0 \leq l(x) \leq l_k$, $x \in X$. x_k is a point where the value of the objective function is small, and its equality and inequality constraints are within the acceptable range. x_k is regarded as a better point, and $l_{k+1} \leq l_k$. Then, under the traditional TR framework, the algorithm considers the minimization subproblem $\min G_k(x)$ with the current iteration point x_k as the center and TR radius $\Delta_k > 0$.

To facilitate the processing of bound constraints, the constraints $\|x - x_k\|_\infty \leq \Delta_k$ and bound constraints of $\min G_k(x)$ are reduced to the corresponding equivalent form: $l \leq x \leq u$, where $t = 1, 2, \ldots, n$.

$$\begin{aligned} l_t &= \max\{\, l_t - x_k^t,\ -\Delta_k \,\}, \\ u_t &= \min\{\, u_t - x_k^t,\ \Delta_k \,\} \end{aligned} \tag{6}$$

Meanwhile, the bound constraint $l \leq x \leq u$ is transformed into the following form.

$$\begin{aligned} g_s &= x_s - l_s \geq 0, \\ g'_s &= u_s - x_s \geq 0 \end{aligned} \tag{7}$$

3.2. **Corrections to the initial generalization of Lagrange multipliers.** The traditional TR algorithm does not make reasonable use of the multiplier information output from the previous iteration when solving the subproblem. To address this deficiency, this paper uses the PTR algorithm to correct the initial multipliers of the subproblem augmented Lagrange function, so as to make the solution model simpler. Firstly, the generalized Lagrange function of $\min G_k(x)$ is given as follows.

$$\begin{aligned} L(x, \eta, \mu, \alpha) = G_k(x) \ - \ \sum_{i=1}^{p} \eta_i\, h_i(x) \ + \ \frac{p}{2} \sum_{i=1}^{p} h_i^2(x) \\ + \ \frac{q}{2\alpha} \sum_{j=1}^{q} \Big(\big(\max\{0,\ \mu_j - \alpha\, g_j(x)\}\big)^2 - \mu_j^2 \Big) \end{aligned} \tag{8}$$

where $\eta \in \mathbb{R}^p$, $\mu \in \mathbb{R}^q$ are augmented Lagrange multipliers and $\alpha \geq 0$ is a penalty factor.

The model is then updated using the minimum F-paradigm method, and the set of interpolated points is updated by changing only one point. At this point, the kth iteration has dropped the objective function sufficiently to replace y_t in the subproblem with x^+:

$$Y_{k+1} \ = \ \big(Y_k \setminus \{y_t\}\big) \ \cup \ \{x^+\} \tag{9}$$

where the PTR algorithm retains the final multipliers η, μ of the above and the initial multipliers η_k, μ_k of the k-th iteration, and at the same time uses η, μ as the initial modified multipliers η_{k+1}, μ_{k+1} of the $(k+1)$-th iteration for solving the augmented generalized Lagrange function of the quadratic model G_{k+1}. Thus, it not only reduces the number of updating iterations of μ, but also makes the function value of the solved sufficiently decreasing and accelerates the convergence speed of the algorithm.

Finally, the set of interpolated points and the model are reconstructed. At this point, Y_k does not satisfy equilibrium, and the objective function of the kth iteration does not decrease sufficiently, even $f(x^+) > f(x_k)$, that is, solving the subproblem yields a function value of x^+ that is larger than the function value of the iteration point, and it is necessary to re-construct the interpolated set of points and the model centred at the current iteration point x_k, such that $x_b = x_k$. Solve the subproblem of the initial multiplier regression initial value $\eta_{k+1} = \eta_0$, $\mu_{k+1} = \mu_0$. The PTR radius Δ_k is updated by calculating the ratio r as shown below.

$$r = \frac{f(x_k) - f(x^+)}{G_k(x_k) - G_k(x^+)} \tag{10}$$

4. Bayesian optimization algorithm based on probabilistic trust region.

4.1. Design of sampling method. Based on the PTR designed above, this paper applies it to the BO algorithm for optimization, using two dynamically varying sizes of PTRs to control the BO algorithm to focus more on local search without losing the ability to optimize globally, with one reliance domain adopted to limit the amount of instances used to train the Gaussian progress during iteration, and the other reliance domain adopted to limit the size of the search space that generates new sampling points. The last step is to use a stochastic search. Finally, a stochastic search is used to optimize the collection function, avoiding the cost of improving collection operation. The suggested PTRBO approach is shown in Figure 3

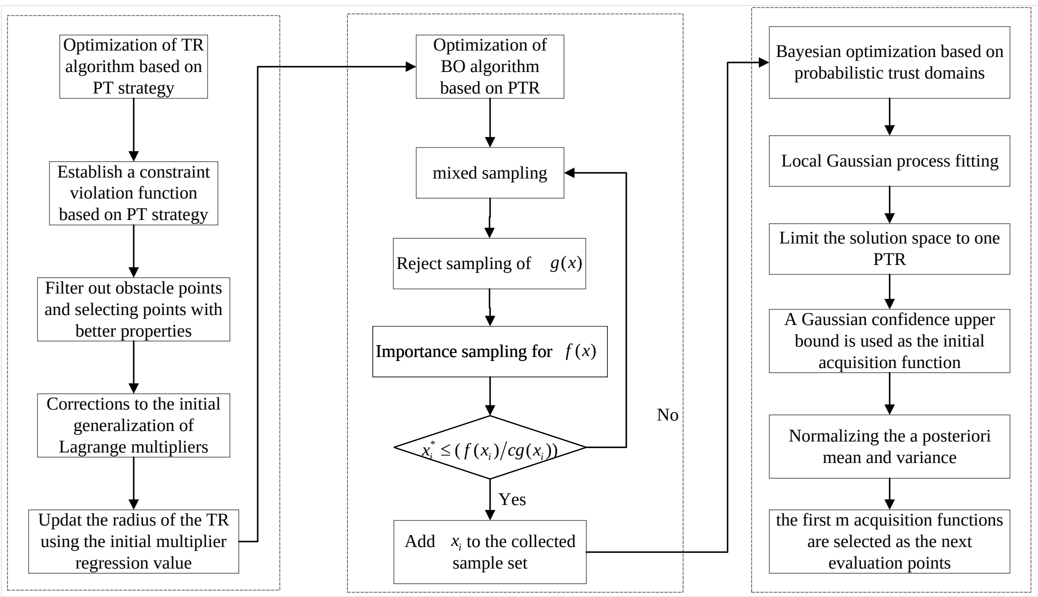

Figure 3. The flow of the PTR optimized BO algorithm

In the Bayesian equation of $P(x \mid y) = \frac{P(y|x)*P(x)}{P(y)}$, the denominator $P(y)$ can be written as $P(y) = \int P(y \mid x) * P(x)\, dx$ when the variables x, y are continuous random variables. Since there are integrals in $P(y)$, there are only a few types of distributions for which $P(y)$

has an analytic solution, and it is therefore difficult to compute integrals with respect to $P(y)$ directly, which is achieved by sampling methods [26].

In this paper, hybrid sampling based on rejection sampling [27] and importance sampling [28] is used to sample distributions that satisfy certain conditions and are easy to sample, which is not only easy to implement, but also obtains high sampling efficiency by extending it to the case of high-dimensional random variables. For any distribution $f(x)$ that needs to be sampled, choose an easy-to-sample distribution $g(x)$ (e.g., normal distribution) such that there exists a constant c, and satisfy the condition $f(x) \leq c\, g(x)$ for any x. Then sample the distribution $g(x)$ and accept the sampling point according to a certain strategy, the main steps are as follows.

(1) Reject sampling of the distribution $g(x)$ to obtain sampling point x_i.

(2) Importance sampling is performed in the interval $(0, 1)$ to obtain the sampling point x^*. Assume that the expectation of the sampling requirement is as follows.

$$E(f) = \int p(x) * f(x)\, dx \tag{11}$$

where $p(x)$ denotes the probability density function of the random variable x. Equation (11) can then be transformed as bellow.

$$E(f) = \int p(x) * f(x)\, dx = \int \frac{p(x) f(x)}{q(x)}\, q(x)\, dx \tag{12}$$

where $q(x)$ denotes the probability density function of the standard normal distribution, such that $Y = \frac{p(x)f(x)}{q(x)}$. Then the above equation is clearly equivalent to finding $E(Y)$. According to the large number theorem, $E(Y)$ can be calculated by Equation (13).

$$E(Y) = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \frac{p(x_i) f(x_i)}{q(x_i)} \tag{13}$$

By taking the corresponding values into the above equation, the expectation of the random variable can be approximated and the original issue can be addressed.

(3) If $x^* \leq \big(f(x_i)/\big(c\, g(x_i)\big)\big)$, then accept x_i. Otherwise, reject. Keep repeating these three steps until getting a certain number of sample points.

4.2. **Bayesian optimization based on probabilistic trust domains.** The performance of the BO algorithm depends on the agent model and the acquisition function, this article will use the PTR algorithm proposed in Section III to optimize the agent model and the acquisition function, so as to improve the operation efficiency of the BO algorithm.

(1) *Optimization of agent models.* In high-dimensional issues, a large amount of sample points are usually needed to exercise the Gaussian progress in order to achieve the ideal fitting effect, which will cause the approach's inefficiency. Motivated from the TR method, this paper abandons the use of Gaussian progress to suit the target operation globally, and instead fits it locally. In the process of optimizing the collection function, the solution space is restricted to a PTR, and a part of the overall data is selected to train the local Gaussian process. Let the current observed data be $D = \{(x_t, y_t)\}_{t=1}^{n}$, the PTR scale parameter be φ, and the current optimal solution be x_{opt}, where n stands for the n-th epoch of the approach, and the set of sample points determined by Equation (14) is used to train the local Gaussian process.

$$D' = \big\{(x, y) : \ \|x - x_{\text{opt}}\| \leq \rho\, \varphi, \ \ (x, y) \in D\big\} \tag{14}$$

where $\rho = \max \lambda_i,\ i \in \{1, 2, \ldots, d\}$, λ_i denote the Gaussian progress' size hyperparameters, and d is the input variable's dimension of the objective function. ρ enables the domain of trust around the partial Gaussian progress to contain the domain of trust around the acquisition operation. The PTR scale parameter φ keeps changing and keeps decreasing during the iteration of the algorithm, so that the proposed local Gaussian process is much less than the global Gaussian process in terms of the number of sample points for training the Gaussian process, which saves a lot of computation time.

(2) *Optimization of the acquisition function.* In order to focus more on local search and global optimization, it is necessary that the next sampled points are generated as close as possible to the current optimal solution. Suppose that the PTR is a hyper-rectangle centered on A and of size B. The length of the edges of each dimension of the hyper-rectangle is as follows.

$$L(i) \ = \ \left(\lambda^{(i)}\varphi\right)^{1/d}\left(\prod_{j=1}^{n}\varphi^{(j)}\right) \tag{15}$$

Then the Gaussian confidence upper bound is used as the initial collection function, and the point set $\{x_i\}_{i=1}^n$ is generated by uniform sampling in the search space $\hat{\Omega}$ of the collection function $a_{\text{ucb}}(x; D_n)$, which is brought into the collection function with the posterior distribution mean $\mu_n(x)$ and variance $\delta_n(x)$ to get the corresponding function values $\{\mu_i\}_{i=1}^n$ and $\{\delta_i\}_{i=1}^n$, respectively, and the posterior mean and variance are normalized using the minimum-maximum normalization to get the mean $\{\mu_i'\}_{i=1}^n$ and the variance $\{\delta_i'\}_{i=1}^n$ to get the final value of the collection function as shown below.

$$a_i \ = \ \mu_i' \ + \ \beta_n\,\delta_i' \tag{16}$$

To produce m candidate points sequentially, $\{a_i\}_{i=1}^n$ is sorted in ascending order, and the point set consisting of x_i related to the first m acquisition operation values a_i is selected as the next evaluation point.

4.3. **Algorithm analysis.** The PTR algorithm proposed in Section 3, which ensures that the solution found is not only globally convergent but also locally exploitable, and the BO algorithm for the PTR improvement described above should also be globally optimizing in nature. Mapping the region of definition of the target operation to the space $\Omega = [0,1]^d$, the optimal search is x_{opt} and the PTR scale parameter is φ.

$$\Omega_{\text{gp}} = \{(x,y) \ | \ \|x_{\text{opt}} - x\|_2 < \varphi, \ (x,y) \in D\} \tag{17}$$

$$\Omega = \{(x^2,y) \ | \ x^{(i)} \in [\ell,u], \ (x^2,y) \in D\} \tag{18}$$

where $\ell^{(i)} = x_{\text{opt}}^{(i)} - \lambda^{(i)}\varphi \Big/ \left(\prod_{j=1}^n \lambda^{(j)}\right)^{1/d}$, $\quad u^{(i)} = x_{\text{opt}}^{(i)} + \lambda^{(i)}\varphi \Big/ \left(\prod_{j=1}^n \lambda^{(j)}\right)^{1/d}$ superscript j denotes the component of the j-th dimension.

Proposition: assume that event $\Omega_{\text{gp}} \neq \emptyset$ is y and event $\{(x_i,y_i) \ | \ (x_i,y_i) \in \Omega_{\text{gp}}, \ (x_i,y_i) \notin \Omega\} \neq \emptyset$ is x. Then there is $\lim_{\varphi\to 0} P(x \mid y) = 0$.

Proof: the PTR of a partial Gaussian progress is a ball focused on x_{opt} with radius $\eta\varphi$. The ball's volume is as follows.

$$V = \frac{\pi^{d/2}(\eta\varphi)^d}{\Gamma(1+d/2)} \tag{19}$$

The PTR around the obtaining operation is a hyper-rectangle with each aspect of $L(i) = \frac{\lambda^{(i)}\varphi}{\left(\prod_{j=1}^n \lambda^{(j)}\right)^{1/d}}$. Since the region of definition of the target operation is $[0,1]^d$, take $L(i) = 1$ when $L(i) \geq 1$, noting that $A = \{1, 2 \ldots, d\}$, is discussed in the following two cases.

(1) The volume of the $L(i) < 1, \ i \in A$, hyperrectangle is φ^d. Assuming that the sampling points not in $\hat{\Omega}_{\text{gp}}$ are consistently allocated in $\hat{\Omega}$, the probability of an observation not in

$\hat{\Omega}_{\mathrm{gp}}$ can be obtained as follows.

$$P(x \mid y) = \frac{V - \varphi^d}{1} = \left(\frac{\pi^{d/2}\eta^d}{\Gamma(1+d/2)} - 1 \right) \varphi^d. \tag{20}$$

Clearly when $\varphi \to 0$, there is $\lim_{\varphi\to 0} P(x \mid y) = \lim_{\varphi\to 0} \left(\frac{\pi^{d/2}\eta^d}{\Gamma(1+d/2)} - 1 \right) \varphi^d = 0$.
(2) Existing $i \in A$ such that $L(i) \geq 1$, if $\forall i \in A$, has $L(i) \geq 1$. Suppose that existing the set $B \neq \emptyset$, $\forall i \in B \subset \{1, 2, \ldots, d\}$, both with $L(i) = 1$, then the magnitude of the hyperrectangle at this point is as follows.

$$V' = \varphi^{|A|-|B|} \; \frac{\prod_{i\in A-B} \lambda^{(i)}}{\left(\prod_{j=1}^{n} \lambda^{(j)}\right)^{1/a}} \tag{21}$$

Let $\zeta = \frac{\prod_{i\in A-B} \lambda^{(i)}}{\left(\prod_{j=1}^{n} \lambda^{(j)}\right)^{1/d}}$, $\quad O = |A| - |B|$, ζ and O be constants, then $P(x \mid y) = \frac{V - \zeta\,\varphi^O}{1} = \frac{\pi^{d/2}(\eta\varphi)^d}{\Gamma(1+d/2)} - \zeta\,\varphi^O$ can be computed to obtain the following equation.

$$\lim_{\varphi\to 0} P(x \mid y) = \lim_{\varphi\to 0} \left(\frac{\pi^{d/2}(\eta\varphi)^d}{\Gamma(1+d/2)} - \zeta\,\varphi^{\alpha} \right) = 0. \tag{22}$$

It follows that when the PTR scale parameter is decreasing, then the optimized BO converges more and more to a primitive BO with a decreasing solution space, making the PTRBO increasingly focused on local exploitation. Conversely when the PTR scale is not so small, this also gives the approach the ability to explore globally as the serach space and the PTR space used to determine the data points of the training local Gaussian process do not exactly overlap.

5. Experiments and analysis of results.

5.1. **Test problem analysis.** This article takes DTLZ, the most widely used test problem set in the field of high-dimensional objective optimization, as an example to estimate the effectiveness of the designed PTRBO approach.To effectively compare the advantages and disadvantages of the algorithms and reduce the interference of random factors in the experiments, this paper uses the evaluation indexes of Inverse Generation Distance (IGD) and Hyper Volume (HV) to evaluate the convergence of the algorithms and the distribution of the problem sets, respectively.

To fairly compare the feasibility of various methods on Bayesian algorithms, comparisons are made with the ACABO method in the literature [11], the TRPBO method in the literature [20], and the TREGO method in the literature [21].The number of decision variables in the DTLZ and their associated parameters are referred to the literature [29]. All relevant algorithms are written in MATLAB R2019a and experimentally simulated on a computer with Intel Core i7 16.00 GHz CPU and 16.0 GB RAM. Each algorithm was run independently 50 times on each test problem and the average of the 50 results was taken as the final result.

Table 1 and Table 2 give the comparative outcome of the IGD and HV metrics obtained by the 4 in the algorithms on the DTLZ{1,2,3} test function when they have 5 objectives and 10 objectives, respectively. The optimal results for each instance in the table are marked in bold. The symbol '+' implies that the comparison method is significantly better than PTRBO; the symbol '-' implies that the comparison method is worse than PTRBO; the symbol '=' indicates that the two algorithms are equally effective.

Table 1. TABLE 1. IGD mean obtained by each algorithm on the DTLZ test suite

Test question	Number of targets	ACABO	TRPBO	TREGO	PTRBO
DTLZ1	5	0.6853 (-)	0.7224 (-)	0.8944 (=)	0.8944
DTLZ1	10	0.8517 (-)	0.8582 (-)	0.8469 (-)	0.9324
DTLZ2	5	0.0095 (-)	0.0109 (=)	0.0118 (+)	0.0109
DTLZ2	10	0.7106 (-)	0.8253 (-)	0.6014 (-)	0.9621
DTLZ3	5	0.5112 (-)	0.6133 (-)	0.5886 (-)	0.6451
DTLZ3	10	0.0042 (-)	0.0061 (-)	0.0074 (-)	0.0079

As can be seen from Table 1 and Table 2, PTRBO achieved 5 optimal IGD results and optimal HV results out of 12 algorithms, and TREGO followed with 2 optimal HV results. PTRBO also has a big advantage in high-dimensional problems, for the DTLZ{1} test problem, which is a linear problem, although the convergence is not as good as that of TRPBO in low- dimensional problems, but the distribution of the optimal set of solutions is more uniform compared to that of ACABO and TREGO. DTLZ{2} is a concave problem and TREGO outperforms PTRBO in dimension 5 due to the fact that TREGO employs randomized weights, where each time the TR is computed it randomly generates weights for all individuals with different objectives leading to a more significant diversity of solution sets and distributional effects for the same number of computations. However, the optimization effect of PTRBO increases significantly when the dimension is increased to 10, which indicates that its global search ability is stronger when the dimension is higher. DTLZ{3} is a degenerate problem, and PTRBO still shows better convergence ability, which indicates that PTRBO is very effective.

Figure 4 demonstrates the discretization of the results achieved by the different algorithms on the DTLZ, the HV metrics reflect the excellent performance of PTRBO in the DTLZ test set, which can take into account both diversity and convergence, with PTRBO having the highest value of HV and a higher median value of HV than the other algorithms. PTRBO not only optimizes the TR algorithm by using the PT strategy, which makes the TR algorithm able to find the local optimal solution quickly, but also further improves the BO by PTR, which enhances the global searching ability, so that the measured performance indexes of PTRBO are better, and the distributability ability is stronger.

5.2. **Performance comparison and analysis.** Given that the PTRBO optimization process is essentially a probabilistic search, this paper compares the performance of different methods using a combination of Gap Metric (GM) and RMSE, and a comparison of the gap metrics of the four different algorithms is shown in Figure 5. GM is an important index to measure the global optimization ability, the larger the GM, the better the optimization effect. From the overall point of view, the GM of the four optimization algorithms have a certain convergence trend, and the degree of GM of PTRBO is higher than the other three algorithms, and when the number of iterations is 80, the GM of ACABO, TRPBO, TREGO and PTRBO are 0.76, 0.75, 0.79, 0.84, respectively, and the best experimental outcome is obtained.

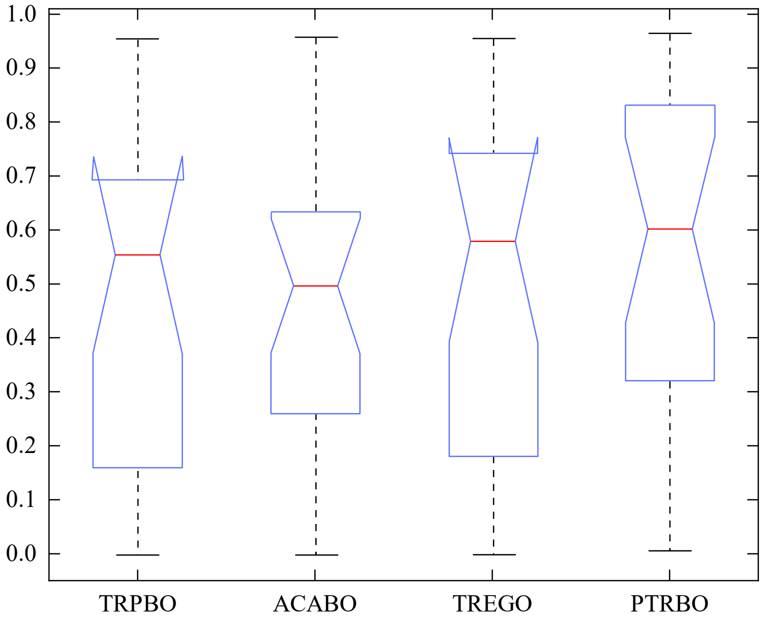

Figure 4. DTLZ test set HV box diagram

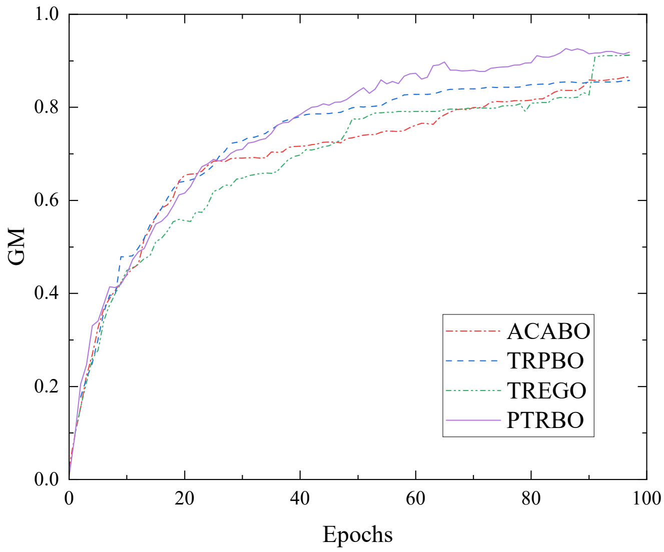

Figure 5. GM for four algorithms

In addition, to verify the effectiveness of PTRBO in a multi-dimensional comparison, RMSE was used for comparative analysis, as shown in Figure 6. From the overall trend, there is a certain convergence trend in the RMSE of the four optimization algorithms, and the RMSE value of PTRBO is 0.23, which is 0.17, 0.19, and 0.07 lower than that of ACABO, TRPBO, and TREGO, respectively. In the early stage of the iteration, the

RMSE result of PTRBO is not the best, but as the number of iteration increases (after about 70 times), PTRBO consistently outperforms the other three algorithms. However, as the number of iterations increases (after about 70 iterations), PTRBO consistently outperforms the other three algorithms. This further implies that the PTRBO has strong optimization ability and robustness.

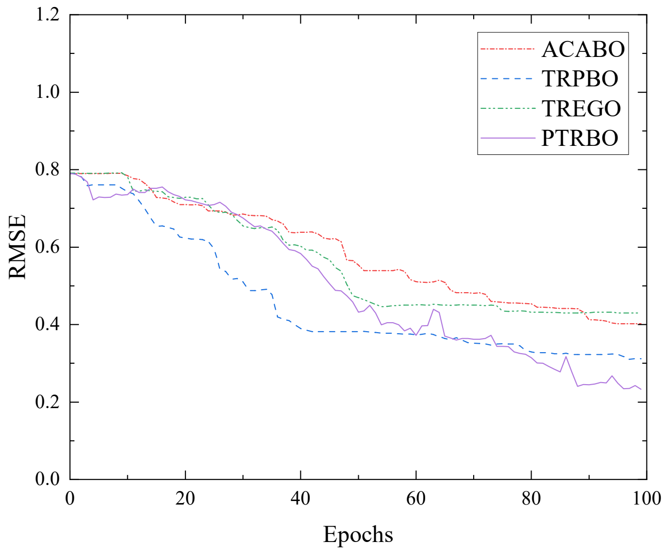

Figure 6. RMSE for four algorithms

6. **Conclusion.** BO is an efficient black-box optimization algorithm that usually works well in cases where the dimensionality of the objective function is not high, but scaling it to higher dimensions is not so easy. In this paper, PTRBO is proposed based on the ideas of probability theory and TR. Firstly, the PTR algorithm was designed to filter the iteration points using the PT strategy before solving the subproblems, and then the initial augmented Lagrange multipliers for solving the subproblems were modified to reduce the number of iterations and iteration time of the algorithm. Two PTRs with the same scale parameter are then used to control the algorithm, allowing the PTRBO to focus on local search and global optimization. One PTR is a ball centered on the current optimal solution with a dynamically changing scale parameter as the radius, and the data for training the Gaussian process are observation points located inside the ball, effectively reducing the number of sample points; the other PTR is a hyper-rectangle around the current optimal solution, and the new evaluation points selected during each iteration are located inside the hyper-rectangle. The experimental results imply that the GM and RMSE of PTRBO are 0.84 and 0.23, respectively, demonstrating strong optimization capability and robustness.

Acknowledgment. This work was supported by the National Science Foundation of the Jiangsu Higher Education Institutions of China (Grant No.21KJB110018), the Qing Lan Project of the Jiangsu Higher Education Institutions 2023 (Grant No.500RCQL23216), the Qing Lan Project of the Nanjing Vocational Institute of Railway Technology 2021 (Grant No.QLXJ202107), the Foundation of Nanjing Vocational Institute of Railway Technology (Grant No.Yr230010), the university student innovation project of Nanjing Vocational Institute of Railway Technology (Grant No.YXKC2023105).

REFERENCES

[1] L. Napalkova, and G. Merkuryeva, "Multi-objective stochastic simulation-based optimisation applied to supply chain planning," Technological and Economic Development of Economy, vol. 18, no. 1, pp. 132-148, 2012.
[2] Y. Zhao, X. Xu, and H. Li, "Minimizing expected cycle time of stochastic customer orders through bounded multi-fidelity simulations," IEEE Transactions on Automation Science and Engineering, vol. 15, no. 4, pp. 1797-1809, 2018.
[3] S. Tao, A. Van Beek, D. W. Apley, and W. Chen, "Multi-model Bayesian optimization for simulation-based design," Journal of Mechanical Design, vol. 143, no. 11, 111701, 2021.
[4] T. Yan, A. Zhou, and S.-L. Shen, "Prediction of long-term water quality using machine learning enhanced by Bayesian optimisation," Environmental Pollution, vol. 318, 120870, 2023.
[5] M. Imani, and S. F. Ghoreishi, "Scalable inverse reinforcement learning through multifidelity Bayesian optimization," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 8, pp. 4125-4132, 2021.
[6] R. E. Haskell, G. Castelino, and B. Mirshab, "Efficient algorithm for locating the global maximum of an arbitrary univariate function," Journal of Forth Application and Research, vol. 5, no. 3, pp. 357-364, 1989.
[7] J. Mockus, "Application of Bayesian approach to numerical methods of global and stochastic optimization," Journal of Global Optimization, vol. 4, pp. 347-365, 1994.
[8] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," Computer Science, vol. 3, no. 6, pp. 23-30, 2010.
[9] R. Moriconi, K. S. Kumar, and M. P. Deisenroth, "High-dimensional Bayesian optimization with projections using quantile Gaussian processes," Optimization Letters, vol. 14, pp. 51-64, 2020.
[10] M. I. Obayya, M. El-Ghandour, and F. Alrowais, "Contactless palm vein authentication using deep learning with Bayesian optimization," IEEE Access, vol. 9, pp. 1940-1957, 2020.
[11] P. Luong, D. Nguyen, S. Gupta, S. Rana, and S. Venkatesh, "Adaptive cost-aware Bayesian optimization," Knowledge-Based Systems, vol. 232, 107481, 2021.
[12] S. Wang, S. H. Ng, and W. B. Haskell, "A multilevel simulation optimization approach for quantile functions," INFORMS Journal on Computing, vol. 34, no. 1, pp. 569-585, 2022.
[13] C.-H. Chen, J. Lin, E. Yücesan, and S. E. Chick, "Simulation budget allocation for further enhancing the efficiency of ordinal optimization," Discrete Event Dynamic Systems, vol. 10, pp. 251-270, 2000.
[14] N. Quan, J. Yin, S. H. Ng, and L. H. Lee, "Simulation optimization via kriging: a sequential search using expected improvement with computing budget constraints," IIE Transactions, vol. 45, no. 7, pp. 763-780, 2013.
[15] G. Pedrielli, S. Wang, and S. H. Ng, "An extended two-stage sequential optimization approach: Properties and performance," European Journal of Operational Research, vol. 287, no. 3, pp. 929-945, 2020.
[16] M. A. Bouhlel, and J. R. Martins, "Gradient-enhanced kriging for high-dimensional problems," Engineering with Computers, vol. 35, no. 1, pp. 157-173, 2019.
[17] Y. Shi, J. Huang, Y. Jiao, and Q. Yang, "A semismooth newton algorithm for high-dimensional nonconvex sparse learning," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 8, pp. 2993-3006, 2019.
[18] A. Ahmed, K. Saleem, O. Khalid, and U. Rashid, "On deep neural network for trust aware cross domain recommendations in E-commerce," Expert Systems with Applications, vol. 174, 114757, 2021.
[19] B. Liu, "A survey on trust modeling from a Bayesian perspective," Wireless Personal Communications, vol. 112, no. 2, pp. 1205-1227, 2020.
[20] J. Zhou, Z. Yang, Y. Si, L. Kang, H. Li, M. Wang, and Z. Zhang, "A trust-region parallel Bayesian optimization method for simulation-driven antenna design," IEEE Transactions on Antennas and Propagation, vol. 69, no. 7, pp. 3966-3981, 2020.
[21] Y. Diouane, V. Picheny, R. L. Riche, and A. S. D. Perrotolo, "TREGO: a trust-region framework for efficient global optimization," Journal of Global Optimization, vol. 86, no. 1, pp. 1-23, 2023.
[22] T.-Y. Wu, H. Li, S. Kumari, and C.-M. Chen, "A Spectral Convolutional Neural Network Model Based on Adaptive Fick's Law for Hyperspectral Image Classification," Computers, Materials & Continua, vol. 79, no. 1, pp. 19-46, 2024.

[23] T.-Y. Wu, A. Shao, and J.-S. Pan, "CTOA: Toward a Chaotic-Based Tumbleweed Optimization Algorithm," Mathematics, vol. 11, no. 10. 2339, 2023.

[24] T.-Y. Wu, H. Li, and S.-C. Chu, "CPPE: An Improved Phasmatodea Population Evolution Algorithm with Chaotic Maps," Mathematics, vol. 11, no. 9. 1977, 2023.

[25] A. Bello, J. Reneses, A. Muñoz, and A. Delgadillo, "Probabilistic forecasting of hourly electricity prices in the medium-term using spatial interpolation techniques," International Journal of Forecasting, vol. 32, no. 3, pp. 966-980, 2016.

[26] A. Blanchard, and T. Sapsis, "Bayesian optimization with output-weighted optimal sampling," Journal of Computational Physics, vol. 425, 109901, 2021.

[27] L. Martino, and J. Míguez, "Generalized rejection sampling schemes and applications in signal processing," Signal Processing, vol. 90, no. 11, pp. 2981-2995, 2010.

[28] D. Csiba, and P. Richtárik, "Importance sampling for minibatches," Journal of Machine Learning Research, vol. 19, no. 27, pp. 1-21, 2018.

[29] L. Yang, K. Li, C. Zeng, S. Liang, B. Zhu, and D. Wang, "Many-objective evolutionary algorithm based on spatial distance and decision vector self-learning," Information Sciences, vol. 624, pp. 94-109, 2023.