

Multibranch Machine Learning-Assisted Optimization and Its Application to Antenna Design

WeiQi Chen, *Student Member, IEEE*, Qi Wu, *Member, IEEE*, Chen Yu, *Member, IEEE*,
Wei Hong, *Fellow, IEEE*, and Haiming Wang, *Member, IEEE*

Abstract—Many full-wave electromagnetic (EM) simulations are needed to design an antenna meeting certain requirements, which involves a considerable computational burden. A multibranch machine learning-assisted optimization (MB-MLAO) method is proposed to dramatically reduce the computational complexity involved in this task. This method is then applied to antenna design and worst-case performance (WCP) searching under a practical manufacturing tolerance. In the conventional Gaussian process regression (GPR)-based MLAO method, a lower confidence bound (LCB) prescreening strategy with an empirical LCB constant is used to weigh the predicted value and predicted uncertainty. Using a variable-fidelity machine learning method, an adaptive LCB variable, and a retraining and repredicting method, the proposed MB-MLAO method can strike a delicate balance between exploitation and exploration in searching. Moreover, variable-fidelity data from full-wave EM simulations are used in the deep GPR machine learning method to further reduce the computational burden. Finally, two test functions and four types of antennas are selected as examples to illustrate the superiority of the proposed MB-MLAO method.

Index Terms—Antenna design, machine learning, optimization, worst-case performance (WCP) searching.

I. INTRODUCTION

WITH the increasing performance specifications of modern antenna systems, especially in the millimeter-wave (mmWave) frequency band, taking the relationships between multiple parameters into account in optimization, sensitivity analysis (SA) and robust design plays an indispensable role in antenna and antenna array design [1]–[3]. Due to the increasing complexities of antenna topology and antenna usage environments, full-wave electromagnetic (EM) simulators are generally used to provide reliable simulation results. However, it is difficult to meet the design targets in a short time through parameter scanning and manual adjustment using a full-wave EM simulator. Moreover, it is almost impossible to obtain analytical solutions for practical antenna optimization problems; therefore, metaheuristic algorithms, for example, genetic

algorithm (GAs) [4], differential evolution algorithms (DEAs) [5], and particle swarm optimization (PSO) [1], [6], have been utilized to solve such problems. Unfortunately, combining a metaheuristic algorithm with a full-wave EM simulator results in another problem; that is, hundreds or thousands of function calls are made before converging to a good result [7]. Each function call requires a time-consuming full-wave simulation, which results in an unbearable computational burden.

An efficient method for reducing the calculation time is training a low-complexity surrogate model using machine learning to replace a very time-consuming EM model. A variety of machine learning-assisted optimization (MLAO) methods have been studied and applied to the field of EM device design and have achieved excellent results, such as artificial neural networks (ANNs) [8], K-nearest neighbor (KNN) algorithms [9], and support vector machines (SVMs) [10]. To further alleviate the computational budget, the variable-fidelity-based MLAO method was introduced [11]–[13]. The low-fidelity (LF) model can be an equivalent-circuit model or a coarse-discretization EM simulation model, and the high-fidelity (HF) model can be a high simulation accuracy EM model. Learning the relationship between data of different fidelities can achieve a higher-precision surrogate model using a small amount of HF data.

Based on the online updating surrogate model mechanism in [14], Gaussian process machine learning (GPML) combined with DEA to assist optimization was applied [15], and using the predicted value and predicted uncertainty provided by the GPML, a lower confidence bound (LCB) prescreening strategy was introduced to balance exploitation and exploration by controlling the LCB constant. The setting of the LCB constant was often based on the designer's experience [13]. The LCB constant was set to 1 in [13], while it was suggested to be equal to 2 in [16]. However, the antenna optimization problem varies case by case, and the setting of the LCB constant strongly impacts the performance and robustness of the algorithm.

In this study, a multibranch machine learning-assisted optimization (MB-MLAO) method is proposed based on the application of adaptive LCB variables and a retraining and repredicting method. In the global optimization algorithm, the fitness function under different LCB variables is used to search through different branches to enhance the robustness of the MLAO algorithm. The data of the LF EM model are simulated to reduce the computational expenditure, and the deep Gaussian process regression (GPR) model is retrained to improve prediction accuracy. Furthermore, the proposed method adaptively verifies one data point by the HF EM model

Manuscript received Aug. 23, 2021; revised 2021; accepted 2021. Date of publication ; date of current . This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1804901, the National Natural Science Foundation of China under Grants 62001101 and 61960206006, and the Science Foundation of Jiangsu Province under Grant BK20200349. (Corresponding author: Haiming Wang.)

The authors are with the State Key Laboratory of Millimeter Waves, Southeast University, Nanjing 210096, China and Purple Mountain Laboratories, Nanjing 211111, China (e-mail: wq_chen@seu.edu.cn; qiwu@seu.edu.cn; chenyu@seu.edu.cn; weihong@seu.edu.cn; hmwang@seu.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier

and adds it to the sample set. The main contributions of this work are as follows.

- 1) The MB-MLAO method provides not only a balance between convergence speed and computational complexity but also excellent adaptation regarding different characteristics and different complexity antenna design problems.
- 2) Based on the surrogate model, the MB-MLAO method is applied using the LCB variable formed by multiple LCB constants to replace a fixed LCB constant to achieve a trade-off between exploration and exploitation, and then multiple converged values and their corresponding input parameters are obtained.
- 3) The LF EM model is calculated to reduce the computational burden, and the surrogate model is retrained. The reprediction values are more accurate because of the existence of corresponding LF data. Based on the reprediction values, one input vector is adaptively selected for the HF EM model to verify. Note that the LCB constant corresponding to each verification value is uncertain, and in each iteration, only one HF EM model is simulated.

The structure of this paper is as follows. Section II briefly explains the mechanism of GPR, the LCB prescreening strategy, and deep GPR and then introduces the MB-MLAO algorithm in detail. Two nonlinear test functions and four practical antennas are used to verify the validation and adaptation of the proposed MB-MLAO in Sections III and IV, respectively. Finally, Section V concludes the paper.

Notations: Upper (Lower) bold-face letters are used to denote matrices (vectors). Superscript $(\cdot)^T$ denotes the transpose. $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with a mean μ and variance σ^2 .

II. MB-MLAO

In this section, GPR, the LCB prescreening strategy and deep GPR are described first. Next, the details of MB-MLAO method are presented.

A. GPR

GPR is a machine learning method based on a Gaussian random process, a kernel trick, and Bayesian inference theory. It has good adaptability for complex problems such as those with high dimensions, small samples, and nonlinearities. GPR is used to obtain a low-cost surrogate model to replace function computation with high complexity. As the GPR example in Fig. 1 shows, the predicted value and predicted uncertainty follow a Gaussian distribution. A brief introduction to GPR is presented as follows. More details of GPR are given in [17], [18]. The training set is $\mathbb{D} = \{\mathbf{X}_{N \times K}, \mathbf{y}_{N \times 1}\}$, $\mathbf{X}_{N \times K} = \{\mathbf{x}^i | i = 1, \dots, N\}$, $\mathbf{x}^i = (x_1^i, x_2^i, \dots, x_K^i) \in \mathbb{R}^K$, $\mathbf{y}_{N \times 1} = \{y^i | i = 1, \dots, N\}$, $y^i \in \mathbb{R}$. N is the number of training samples, and K is the dimension of the input vector. The GPR model is treated as

$$y^i = f(\mathbf{x}^i) + \epsilon^i, \quad (1)$$

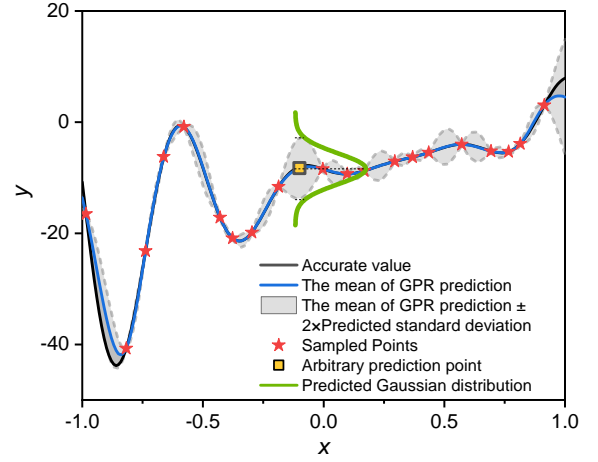


Fig. 1. An illustrated example of predicted values and predicted uncertainties using GPR.

where $f(\mathbf{x}^i)$ is a regression term and ϵ^i is an independent error term that follows the Gaussian distribution. Assume that

- 1) $y^i = \mu + \epsilon(\mathbf{x}^i)$, $\epsilon(\mathbf{x}^i) \sim \mathcal{N}(0, \sigma^2)$.
- 2) The correlation of any two points, $\epsilon(\mathbf{x}^i)$ and $\epsilon(\mathbf{x}^j)$, is dependent on the distance between the corresponding points, which is described as

$$\text{corr}(\epsilon(\mathbf{x}^i), \epsilon(\mathbf{x}^j)) = \exp\left(-\sum_{q=1}^K \theta_q |x_q^i - x_q^j|^{p_q}\right), \quad (2)$$

where μ , σ^2 , $\boldsymbol{\theta} = \{\theta_q | \theta_q \geq 0, q = 1, \dots, K\}$ and $\mathbf{p} = \{p_q | 1 \leq p_q \leq 2, q = 1, \dots, K\}$ are hyperparameters that need to be estimated.

The likelihood function is given by

$$\mathcal{L}(\mu, \sigma^2, \boldsymbol{\theta}, \mathbf{p}) = \frac{\exp\left(-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{1}\mu)^T \mathbf{C}^{-1}(\mathbf{y} - \mathbf{1}\mu)\right)}{(2\pi\sigma^2)^{\frac{N}{2}} |\det(\mathbf{C})|^{\frac{1}{2}}}, \quad (3)$$

where $\mathbf{1}$ denotes an N -dimensional column vector of ones and \mathbf{C} is a correlation matrix of size $N \times N$. Two hyperparameters μ and σ are estimated by using the maximum likelihood criterion for (3)

$$\hat{\mu} = \frac{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{y}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}}, \quad (4)$$

$$\hat{\sigma}^2 = \frac{(\mathbf{y} - \mathbf{1}\hat{\mu})^T \mathbf{C}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu})}{N}. \quad (5)$$

Then, $\boldsymbol{\theta}$ and \mathbf{p} can be estimated by maximizing (3). For test point \mathbf{x}^* , the best linear unbiased predictor is

$$\tilde{y}(\mathbf{x}^*) = \hat{\mu} + \mathbf{r}^T \mathbf{C}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}), \quad (6)$$

$$s^2(\mathbf{x}^*) = \hat{\sigma}^2 \left[1 - \mathbf{r}^T \mathbf{C}^{-1} \mathbf{r} + \frac{(1 - \mathbf{1}^T \mathbf{C}^{-1} \mathbf{r})^2}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \right], \quad (7)$$

where $\mathbf{r}^T = [\text{corr}(\epsilon(\mathbf{x}^*), \epsilon(\mathbf{x}^1)), \dots, \text{corr}(\epsilon(\mathbf{x}^*), \epsilon(\mathbf{x}^N))]$, $\tilde{y}(\mathbf{x}^*)$ is the prediction mean, and $s^2(\mathbf{x}^*)$ is the prediction uncertainty.

B. LCB prescreening strategy

In the LCB prescreening strategy [19], [20], the prediction value and uncertainty given by (6) and (7) are combined as

$$y_{\text{LCB}}(\mathbf{x}^*) = \tilde{y}(\mathbf{x}^*) - \omega s(\mathbf{x}^*), \omega \in [0, 3], \quad (8)$$

where $s(\mathbf{x}^*)$ is the square root of $s^2(\mathbf{x}^*)$. A qualitative analysis is given that balances the exploration and exploitation of the LCB constant ω . More details are shown in [17].

Let us consider two extreme cases. In case I, the test point \mathbf{x}^* is very far from every observed point, $\mathbf{r} \approx \mathbf{0}$, and s is approximately σ . In case II, the test point \mathbf{x}^* is the same as one observed point. Let $\mathbf{x}^* = \mathbf{x}^i$ and $\mathbf{C}^{-1}\mathbf{r} = \mathbf{e}_i$, where \mathbf{e}_i is the i -th unit vector. The second item of (7) is expressed by

$$\mathbf{r}^T \mathbf{C}^{-1} \mathbf{r} = \mathbf{r}^T \mathbf{e}_i \equiv \text{corr}(\epsilon(\mathbf{x}^i), \epsilon(\mathbf{x}^i)) = 1, \quad (9)$$

and the last item is given by

$$\mathbf{1}^T \mathbf{C}^{-1} \mathbf{r} = \mathbf{1}^T \mathbf{e}_i = 1. \quad (10)$$

Then, we have $s = 0$. The size of s depends on the distance between the test point and the observed points, that is, the correlation between them. The LCB prescreening strategy is used in minimization problems; that is, when ω is small, the optimal solution is more likely to be the most promising candidate points, and when ω is large, the searching path trends toward unexplored areas, where the observed points are sparse.

C. Deep GPR

The HF model yields accurate simulation results but is time-consuming, and the LF model has a short simulation time but low accuracy. To learn the relationships between different fidelities, a small amount of HF data is used to train the surrogate model to reduce the computational burden. For multifidelity models, a recursive Gaussian process can be used to establish a regression surrogate model. The fidelity training set of T levels is organized as $\mathbb{D} = \{\mathbf{x}_t, y_t\}, t = 1, \dots, T$, and the subscript from small to large represents the fidelity level from low to high. \mathbf{x}_t and y_t represent the input and output of the training set, respectively, and y_t denotes the simulation data of the t -level fidelity model. The autoregressive model of [21] is

$$f_t(\mathbf{x}) = \rho_{t-1} f_{t-1}(\mathbf{x}) + \delta_t(\mathbf{x}), 2 \leq t \leq T, \quad (11)$$

where f_{t-1} and f_t are the regression items at the fidelity levels $t-1$ and t , respectively, ρ_{t-1} is the correlation factor between them and $\delta_t \sim \mathcal{GP}(\mu_{\delta_t}, k_t(\mathbf{x}_t^i, \mathbf{x}_t^j; \theta_t))$. k_t is the covariance function. The mean μ_{δ_t} and hyperparameter θ_t need to be estimated. Inspired by deep learning [22], we generalize (11) to

$$f_t(\mathbf{x}) = g_t(\mathbf{x}, f_{t-1}(\mathbf{x})), \quad (12)$$

where g_t is a function that maps the LF model to the HF model following a Gaussian distribution, $g_t \sim \mathcal{GP}(\mathbf{0}, k_t((\mathbf{x}^i, f_{t-1}(\mathbf{x}^i)), (\mathbf{x}^j, f_{t-1}(\mathbf{x}^j)); \theta_t))$, and the GP prior f_{t-1} is replaced by the GP posterior f_{t-1}^* .

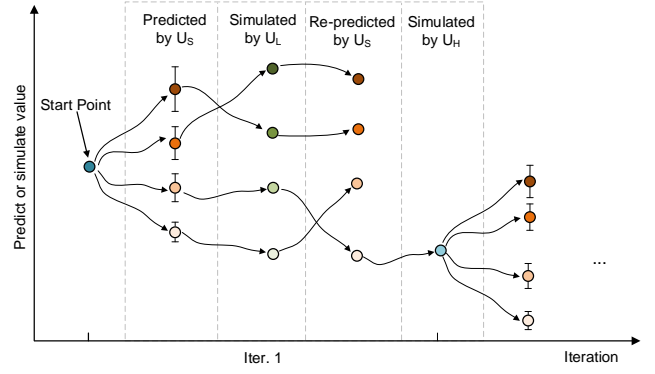


Fig. 2. Schematic diagram of the proposed MB-MLAO method.

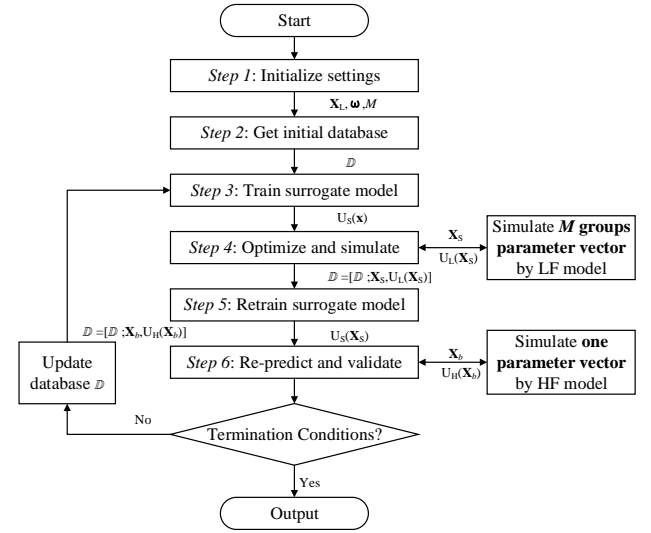


Fig. 3. Flow diagram of the proposed MB-MLAO method.

D. MB-MLAO framework

The schematic of the proposed MB-MLAO method is shown in Fig. 2, where U_L , U_H and U_S represent the LF EM simulation model, HF EM simulation model, and surrogate model, respectively. The flow diagram is shown in Fig. 3, and the steps of the MB-MLAO procedure are shown as follows:

Step 1. Initialize settings:

Define the optimized parameters, optimized specifications, search space $[a, b]^K$, the number of LCB constants M and their values $\omega = \omega_i, i = 1, \dots, M$. N_L parameter vectors are sampled by Latin hypercube sampling (LHS) [23] within the searching space, denoted as \mathbf{X}_L .

Step 2. Obtain the initial database:

Calculate \mathbf{X}_L by LF EM simulation, and obtain the response $\mathbf{R}_L = U_L(\mathbf{X}_L)$. Select N_H parameter vectors from \mathbf{X}_L , denoted as \mathbf{X}_H , calculate \mathbf{X}_H by HF EM simulation, and obtain $\mathbf{R}_H = U_H(\mathbf{X}_H)$. The initial database is

$$\mathbb{D} = \begin{bmatrix} \mathbf{X}_L & \mathbf{R}_L \\ \mathbf{X}_H & \mathbf{R}_H \end{bmatrix}. \quad (13)$$

Step 3. Train the surrogate model:

Use deep GPR learning for the relationship between the input parameters and response to obtain a low-cost surrogate model U_S .

Step 4. Optimize based on the surrogate model:

Use a global algorithm such as a GA to optimize the surrogate model constructed in *Step 3*. The fitness function can be expressed as

$$F(\mathbf{x}_i) = \min_{\mathbf{x}_i} \{\tilde{y}(\mathbf{x}_i) - \omega_i \hat{s}(\mathbf{x}_i)\}, i = 1, \dots, M. \quad (14)$$

Obtain the optimal values and input parameters $\mathbf{X}_S = \{\mathbf{x}_i | i = 1, \dots, M\}$ under M groups of different branches.

Step 5. Simulate and retrain the surrogate model:

To control the additional computational budget and to improve the prediction accuracy, use the LF EM simulation to calculate the above M groups of input parameters \mathbf{X}_S . Then, add the M groups LF data into the database \mathbb{D} , and retrain the surrogate model.

Step 6. Repredict and validate:

By repredicting the M group input parameters, due to the existence of LF data, the prediction accuracy of the M group input parameters is improved. Select the input parameter vector corresponding to the optimal repredicted value for the HF EM model to verify. Note that the validated input parameter vector is the algorithm adaptively selected from M groups of parameters. If the termination condition, such as the optimized target or the maximum number of iterations, is met, stop the loop; otherwise, update the database, and repeat *Step 3*.

III. VERIFICATION USING TEST FUNCTIONS

To illustrate the superiority of the MB-MLAO algorithm over the conventional single-branch MLAO (SB-MLAO) algorithm, two test functions are listed as examples. The function libraries of emukit [24] and geatpy [25] based on Python are used here.

The K -dimensional, multimodal and variable-fidelity test problem Ackley (Function I) and 5-dimensional, unimodal and variable-fidelity test problem Ellipsoid (Function II) [26] are used here. Function I is defined by

$$\begin{cases} y_H = -20 \exp \left\{ -\frac{1}{5} \sqrt{\frac{1}{K} \sum_{i=1}^K (x_i - a_i)^2} \right\} \\ \quad - \exp \left\{ \frac{1}{K} \sum_{i=1}^K \cos(2b\pi x_i - a_i) \right\} + 20 + \exp \{1\} \\ y_L = -20 \exp \left\{ -\frac{1}{5} \sqrt{\frac{1}{K} \sum_{i=1}^K x_i^2} \right\} \\ \quad - \exp \left\{ \frac{1}{K} \sum_{i=1}^K \cos 2\pi x_i \right\} + 20 + \exp \{1\} \end{cases} \quad (15)$$

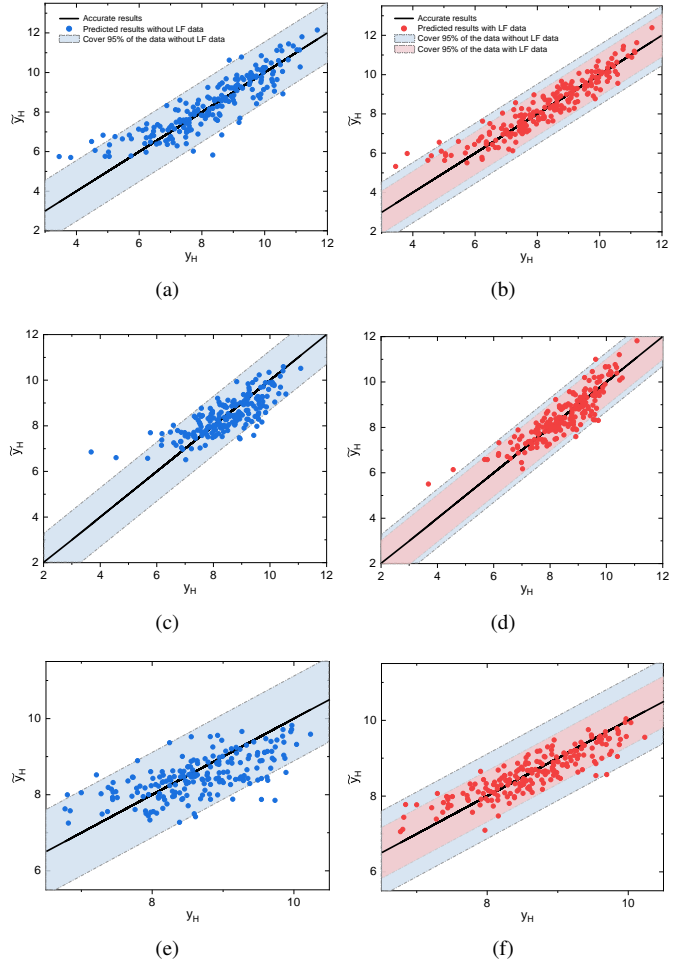


Fig. 4. The exact and predicted values of Function I. (a) $K=5$, without LF data; (b) $K=5$, with LF data; (c) $K=10$, without LF data; (d) $K=10$, with LF data; (e) $K=20$, without LF data; (f) $K=20$, with LF data.

TABLE I
THE SETTINGS OF FUNCTION I AND MMSE WITHIN 200 TESTS.

Dimension and Search Range	N_H	N_L	MMSE	
			Case I	Case II
$[-3, 3]^5$	20	100 (101)	0.53	0.31
$[-3, 3]^{10}$	20	100 (101)	0.49	0.28
$[-3, 3]^{20}$	50	100 (101)	0.33	0.13

where $K = \{5, 10, 20\}$, $b = 1.3$ and $\mathbf{a} = [1.2, 0.2, 1.4, 0.8, 1.8, 1.0, 1.6, 0.6, 2.0, 0.4, 1.3, 0.3, 1.5, 0.9, 1.9, 1.1, 1.7, 0.7, 2.1, 0.5]$. In addition, Function II is expressed as

$$\begin{cases} y_H = \sum_{i=1}^5 ia_i(x_i - b_i)^2 \\ y_L = \sum_{i=1}^5 ix_i^2 \end{cases} \quad (16)$$

where $\mathbf{a} = [0.3, 0.4, 0.2, 0.6, 1.0]$ and $\mathbf{b} = [1.8, 0.4, 2.0, 1.2, 1.4]$.

To compare the differences between the predicted data with and without the corresponding LF data, Function I with dimensions of 5, 10, and 20 is tested. The numbers of HF

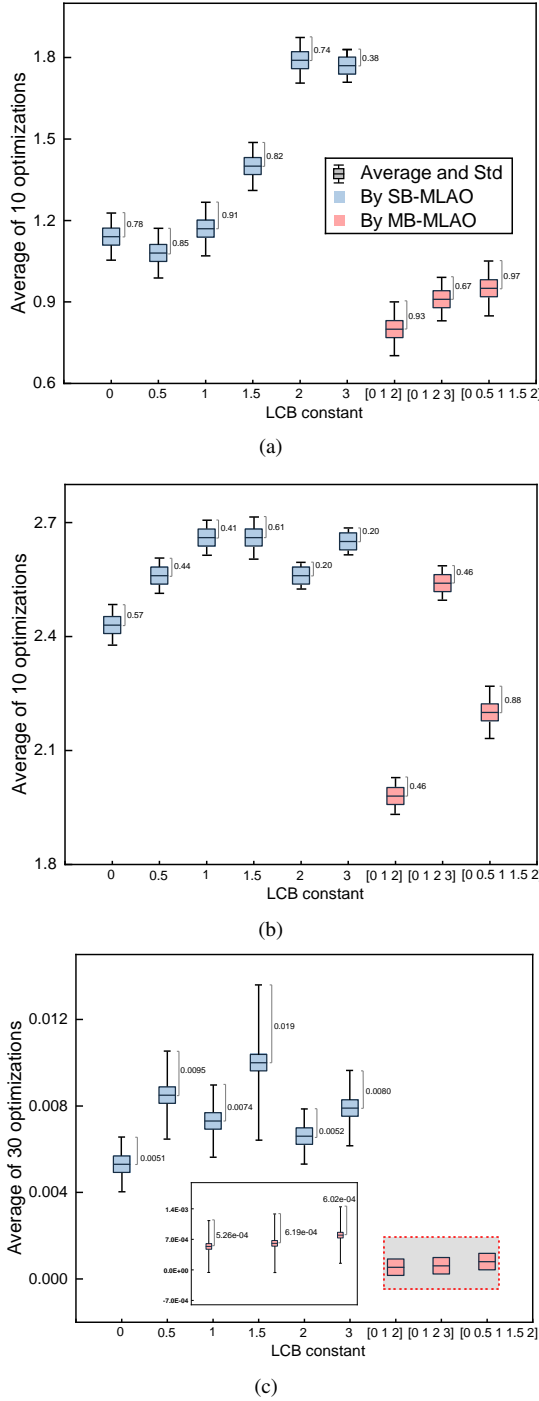


Fig. 5. Average and standard deviation (Std) of test functions. (a) Function I with $[-3, 3]^5$; (b) Function I with $[-3, 3]^10$; (c) Function II with $[-3, 3]^5$

and LF training sets are shown in Table I. After the surrogate model is trained, case I indicates that the test data are directly predicted. Case II indicates that the test data are first calculated using the LF model, and the LF data are added to the training set for retraining; then, the test data are predicted. There are 200 sets of test data. Note that for each set of test data, case II has only one more LF data point than case I. In Fig. 4, the blue dots and red dots represent case I and case II, respectively. The blue shade and red shade represent the area covering 95% of the predicted points for 200 sets of test data for case I and

case II, respectively. The red dots are more concentrated on the black line in the middle than the blue dots. In Table I, after retraining, the mean of the mean square error (MMSE) is smaller, which means that the accuracy of the prediction is improved.

We assume that the calculation time of the HF model τ_{HF} , training time τ_{train} , optimization time τ_{opti} , and calculation time of the LF model τ_{LF} follow the relationships

$$\tau_{\text{HF}} = 20\tau_{\text{LF}}, \tau_{\text{train}} = 0.2\tau_{\text{LF}}, \text{ and } \tau_{\text{opti}} = 0.5\tau_{\text{LF}}. \quad (17)$$

The total optimization time τ_{total} can be expressed as

$$\tau_{\text{total}} = \tau_{\text{initial}} + I(M\tau_{\text{LF}} + \tau_{\text{HF}} + \rho\tau_{\text{train}} + M\tau_{\text{opti}}), \quad (18)$$

where

$$\rho = \begin{cases} 1, & \text{SB-MLAO,} \\ 2, & \text{MB-MLAO,} \end{cases}$$

τ_{initial} represents the time to prepare the initial database, M is the number of LCB variables, and I is the number of iterations of MLAO. For SB-MLAO, M equals 1, and $\tau_{\text{total},S} = \tau_{\text{initial}} + 21.7I_S\tau_{\text{LF}}$. For MB-MLAO, $\tau_{\text{total},M} = \tau_{\text{initial}} + (20.4 + 1.5M)I_M\tau_{\text{LF}}$. In the case of $I_S = 100$, $\tau_{\text{total},S} \geq \tau_{\text{total},M}$ can be obtained if $I_{M=3} \leq 87$, $I_{M=4} \leq 82$ and $I_{M=5} \leq 77$. When $I_S = 35$, the condition $I_{M=3} \leq 30$, $I_{M=4} \leq 28$ and $I_{M=5} \leq 27$ should be satisfied such that $\tau_{\text{total},S} \geq \tau_{\text{total},M}$.

The results of MB-MLAO and SB-MLAO are exhibited in Fig. 5. For Function I and Function II, 10 and 30 independent runs are performed, respectively. As shown in Fig. 5(b), the worst performance using the proposed algorithm with $\omega = [0, 1, 2, 3]$ falls back to that using SB-MLAO. Even in this group, the average minimum value using MB-MLAO in 10 iterations is still better than that using SB-MLAO with $\omega = 1, 1.5$, or 3. In other tests, it is inferred that for problems with different features and different complexities, MB-MLAO has better optimization efficiency and robustness than SB-MLAO.

IV. APPLICATION TO ANTENNA DESIGN

To verify the superiority of the proposed algorithm, four antennas are listed as examples. The input parameters before and after optimization are shown in Appendix A.

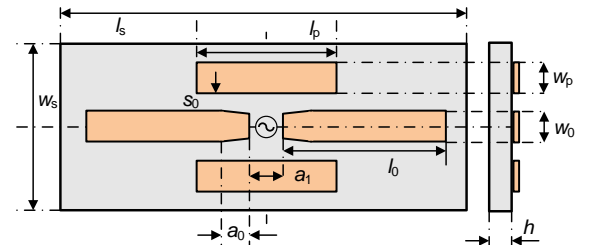


Fig. 6. The geometry of the UWB antenna (Antenna 1) [27].

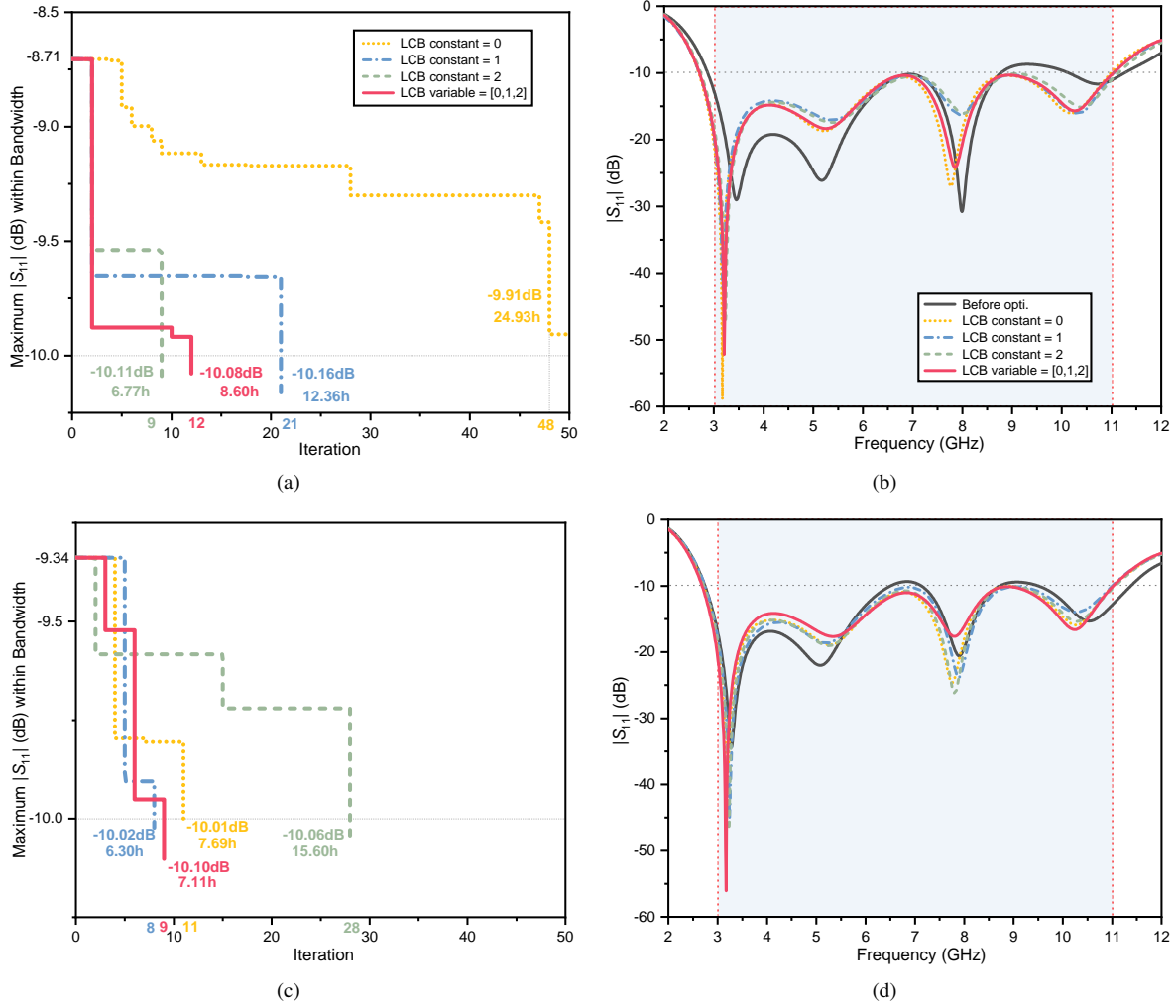


Fig. 7. Convergence curves and the reflection coefficients for Antenna 1 using different methods. (a) and (b) Initial database 1; (c) and (d) Initial database 2.

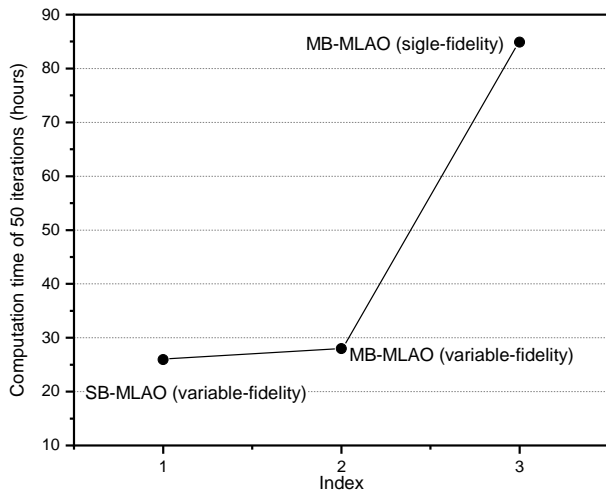


Fig. 8. Estimated computational times for the UWB antenna with 50 iterations based on different strategies.

A. UWB planar dipole antenna (Antenna 1)

The ultrawideband UWB antenna proposed in [27] is fabricated with a Rogers 5880 substrate of relative permittivity

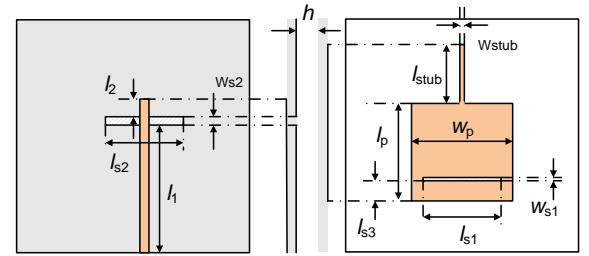


Fig. 9. The geometry of the triband patch antenna (Antenna 2) [28].

$\epsilon_r = 2.2$ and with $h = 1.58$ mm. Its geometry is shown in Fig. 6. The design parameters are $[l_0, w_0, a_0, l_p, w_p, s_0]$. The low bounds are $[18, 12, 0.3, 12, 5, 0.8]$ mm, and the high bounds are $[20, 14, 0.7, 14, 7, 1.2]$ mm.

The HF model of the UWB antenna is simulated through CST Microwave Studio [29] with approximately 9.6 million mesh cells and an evaluation time of 1600 seconds. The LF model uses approximately 126 thousand mesh cells with a computational time of 43 seconds. The optimization goal is to maximize $|S_{11}|$ to less than -10 dB within 3 – 11 GHz. The LF training data input parameter vector \mathbf{X}_L is sampled by the

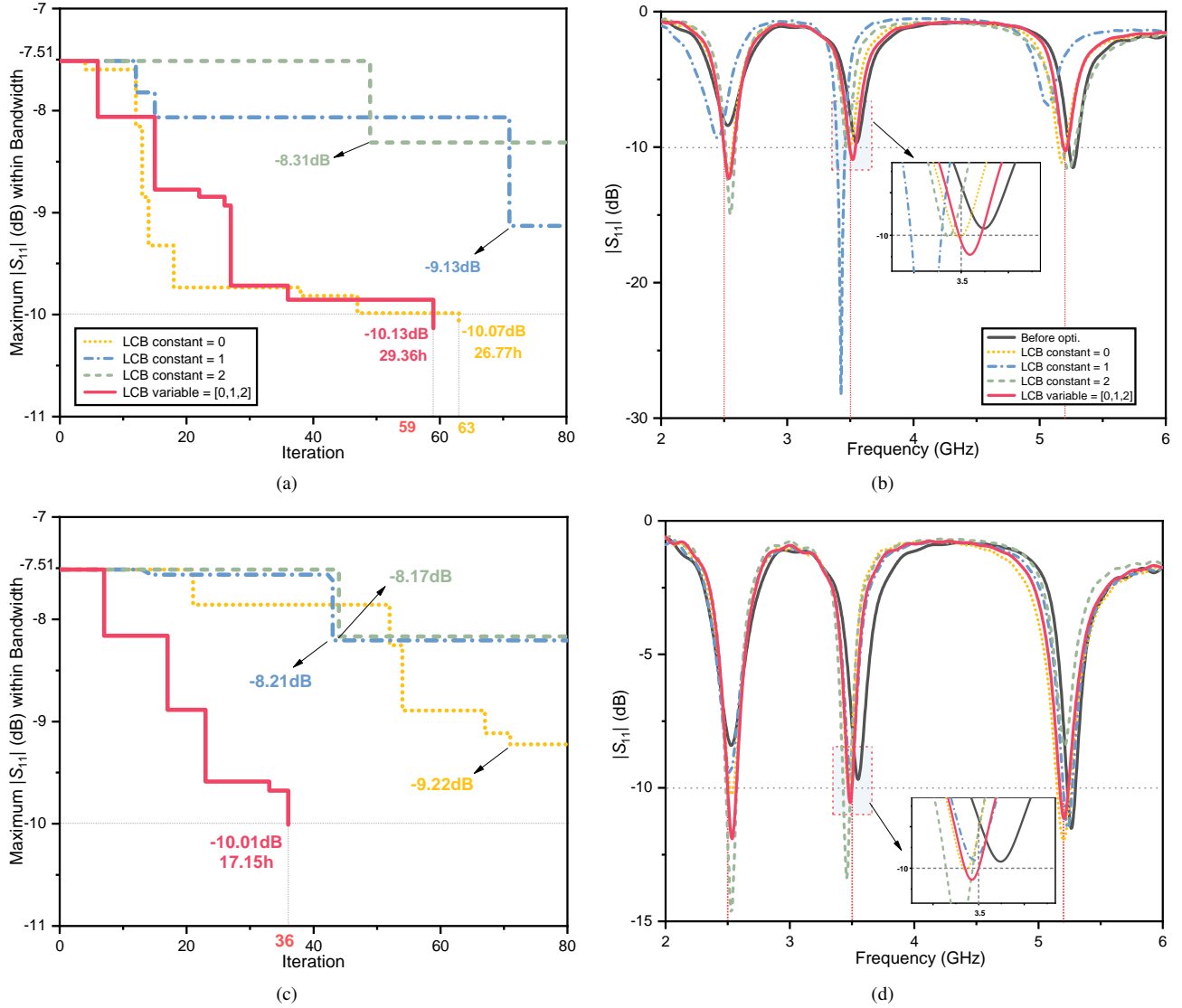


Fig. 10. Convergence curves and the reflection coefficients of Antenna 2 using different methods. (a) and (b) $N_L = 100$, $N_H = 20$. (c) and (d) $N_L = 50$, $N_H = 10$.

LHS method with $N_L = 40$, and HF training data are selected from \mathbf{X}_L with $N_H = 5$. The maximum within bandwidths of $|S_{11,i}^L|$, $i = 1, \dots, N_L$ and $|S_{11,i}^H|$, $i = 1, \dots, N_H$ are used as the output of the surrogate model. The fitness function can be described as

$$f(\mathbf{x}_i) = \min_{\mathbf{x}_i} \{ \tilde{R}_S(\mathbf{x}_i) - \omega_i R_S^s(\mathbf{x}_i) \}, \quad i = 1, 2, \dots, M, \quad (19)$$

where $\tilde{R}_S(\mathbf{x}_i)$ and $R_S^s(\mathbf{x}_i)$ represent the predicted value and predicted uncertainty, respectively. The termination condition is reaching the optimization goal or 50 iterations. In the MB-MLAO method, multiple LCB variables are denoted as $\omega = [0, 1, 2]$, and they are compared with the LCB constant in the SB-MLAO method in which ω equals 0, 1, or 2, respectively. Figs. 15(a) and 15(b) show the changes in the LCB variables of the corresponding HF EM simulation in the iteration of the MB-MLAO method. Fig. 7 gives the convergence curves of two initial sample sets and their S_{11} . In Fig. 7(a), the optimal LCB constant is $\omega = 2$. However, in Fig. 7(c), the case of $\omega = 2$ is the worst. In this example, the

proposed MB-MLAO method can achieve the goal within 9 hours in both tests and is more robust than SB-MLAO. The estimated computation times of different strategies are given in Fig. 8. Single-fidelity multibranch optimization means setting multiple LCB constants in the optimization process, but HF models are utilized during the modeling and verification stages without the need for retraining and repredicting. Multifidelity verification, retraining, and repredicting greatly shorten the calculation time.

B. Triband patch antenna (Antenna 2)

A triband antenna is utilized to verify the MB-MLAO method. The geometry of the triantenna is shown in Fig. 9, and more design details are given in [28]. The design parameters are $[l_2, l_{s1}, l_{s2}, l_{s3}, w_{s1}, w_{s2}, l_p, w_p, l_{stub}]$. The low bounds are [5, 19, 20, 5, 0.2, 1.5, 32, 31, 16] mm, and the high bounds are [9, 26, 25, 7.5, 2.2, 2.5, 35, 34, 20] mm.

The HF simulation model has approximately 2 million mesh cells, and the computation time is 960 seconds. The LF

simulation model uses approximately 109 thousand mesh cells, with a computational time of 94 seconds. The optimization goal is for the $|S_{11}|$ values of 2.5 GHz, 3.5 GHz, and 5.2 GHz to be less than -10 dB. The output vector is the $|S_{11}|$ of the three frequency points, and the three frequency points are added to the training database as a feature dimension. The optimization problem can be converted such that the maximum of the three frequency points is less than -10 dB, expressed as

$$f(\mathbf{x}_i) = \min_{\mathbf{x}_i} \{ \max_{f_j} \{ \tilde{R}_S(\mathbf{x}_i, f_j) - \omega_i R_S^s(\mathbf{x}_i, f_j) \} \}, \quad (20)$$

where $f_j \in [2.5, 3.5, 5.2]$, $j = 1, 2, 3$, and $i = 1, \dots, M$. $\tilde{R}_S(\mathbf{x}_i, f_j)$ represents the predicted value of \mathbf{x}_i at f_j . Fig. 15(c) and Fig. 15(d) are the changes in LCB variables of the corresponding HF EM simulation. Fig. 10 gives some typical results. In Fig. 10(a), $N_L = 100$, $N_H = 20$, and the necessary time to generate the initial sample is 7.94 hours. Only $\omega = 0$ and $\omega = [0, 1, 2]$ reach the optimization target within the specified number of 80 iterations. In Fig. 10(c), $N_L = 50$, $N_H = 10$, and only the proposed MB-MLAO method reaches the goal, taking 17.15 hours. The proposed method is more robust than the SB-MLAO method when the amounts of HF and LF data are different for the same antenna.

C. Dual-band monopole antenna (Antenna 3)

A dual-band monopole antenna for wireless local area network (WLAN) application is used as an example. As the geometry in Fig. 11 shows, Antenna 3 is designed based on a single-layer substrate with height $h = 1$ mm and a relative permittivity of $\epsilon_r = 4.4$, and those monopoles are printed on both sides of the substrate. The patch on the front is similar to the patch on the back but with two more L-shaped branches. The design parameters are $[l_1, l_2, l_3, l_4, l_{s1}, l_{s2}, w_3, w_{s1}]$. The low bounds are $[2.4, 0.7, 1.8, 12, 15.5, 0.5, 1.1, 0.5]$ mm, and the high bounds are $[2.8, 0.9, 2.2, 13, 16.5, 0.8, 1.3, 0.8]$ mm. The other parameters shown in Fig. 11 are $w_1 = 3.87$ mm and $w_2 = 9.67$ mm.

$N_H = 6$, and $N_L = 30$. Different from Antenna 2, which requires the optimization of the response of three frequency points, this example requires the optimization of the response of two frequency bands. The number of mesh cells of the HF model is approximately 5.26 million, and the evaluation time is 0.65 hours. The LF model uses approximately 740 thousand mesh cells, with a computational time of 218 seconds. The optimization goal is for the $|S_{11}|$ values from 2.4 GHz to

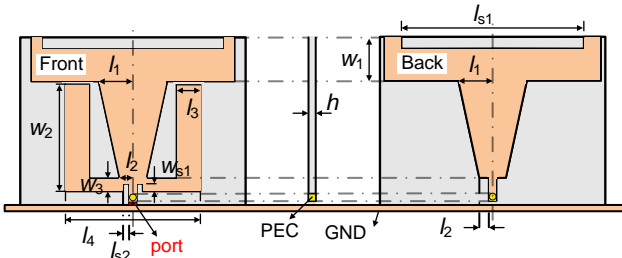


Fig. 11. The geometry of the dual-band antenna (Antenna 3).

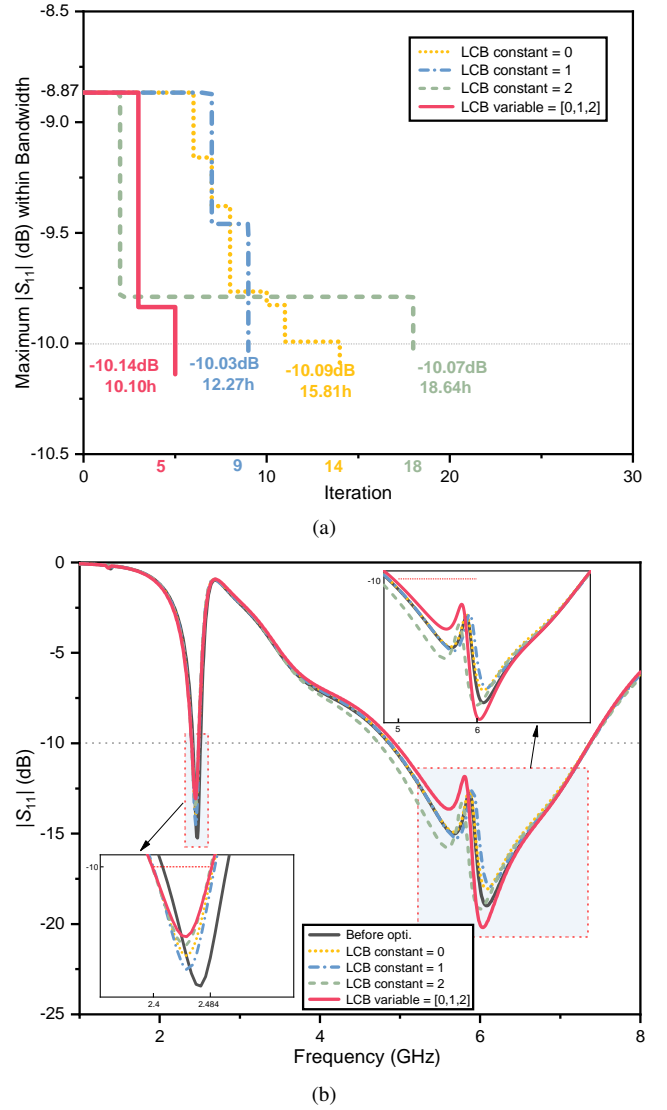


Fig. 12. Convergence curves and reflecting coefficients of Antenna 3 using different methods.

2.484 GHz and 5 GHz to 6 GHz to be less than -10 dB. The output vector of the training data is the maximum of the $|S_{11}|$ of the two bands. The optimization problem is formulated as

$$f(\mathbf{x}_i) = \min_{\mathbf{x}_i} \{ \tilde{R}_S(\mathbf{x}_i) - \omega_i R_S^s(\mathbf{x}_i) \}, i = 1, \dots, M. \quad (21)$$

The termination condition is reaching the optimization goal or a maximum iterations of 30. The necessary time to generate the initial sample is 5.69 hours. Fig. 15(e) shows the changes in the LCB variables of the corresponding HF EM simulation. The comparison result is shown in Fig. 12. The proposed algorithm is the fastest to reach the goal, and the optimization is completed in 5 iterations.

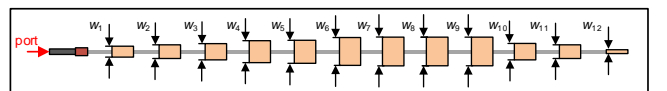


Fig. 13. The geometry of a series-fed microstrip array antenna (Antenna 4).

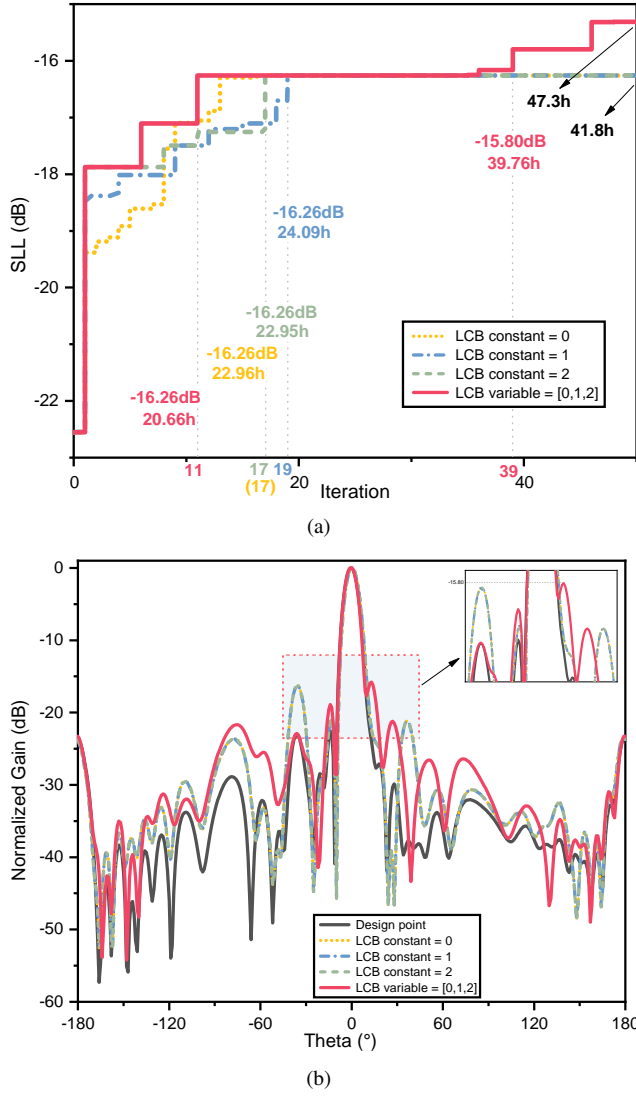


Fig. 14. The WCP searching convergence and normalized patterns of Antenna 4 using different methods.

D. Series-fed microstrip array antenna (Antenna 4)

Antenna 4 is fabricated on a single-layer Rogers RO3003 and is designed for 77 GHz automotive radar applications with 12 series-fed microstrip antenna elements [30]. Its worst-case performance (WCP) is determined using the proposed MB-MLAO. In the 77 GHz band, the size of the antenna is smaller, and the manufacturing tolerance requirement is stricter. WCP searching is a popular method for analyzing the sensitivity of antennas when manufacturing tolerances are known prior [1]. The geometry of Antenna 4 is shown in Fig. 13. The design parameters are $[w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}, w_{11}, w_{12}]$. The design points are $\mathbf{x}_0 = [0.56, 0.7, 0.84, 1.12, 1.19, 1.4, 1.47, 1.47, 0.84, 0.7, 0.21]^T$ mm. The input tolerances are $\delta = [0.1, 0.1, 0.1, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.1, 0.1, 0.05]^T$ mm. The searching range is $\mathbf{x}_0 \pm \delta$. The HF responses are simulated in CST Microwave Studio with approximately 5 million mesh cells, and the computation time is 0.52 hours. The LF responses are simulated with approximately 201 thousand mesh cells, and the computation

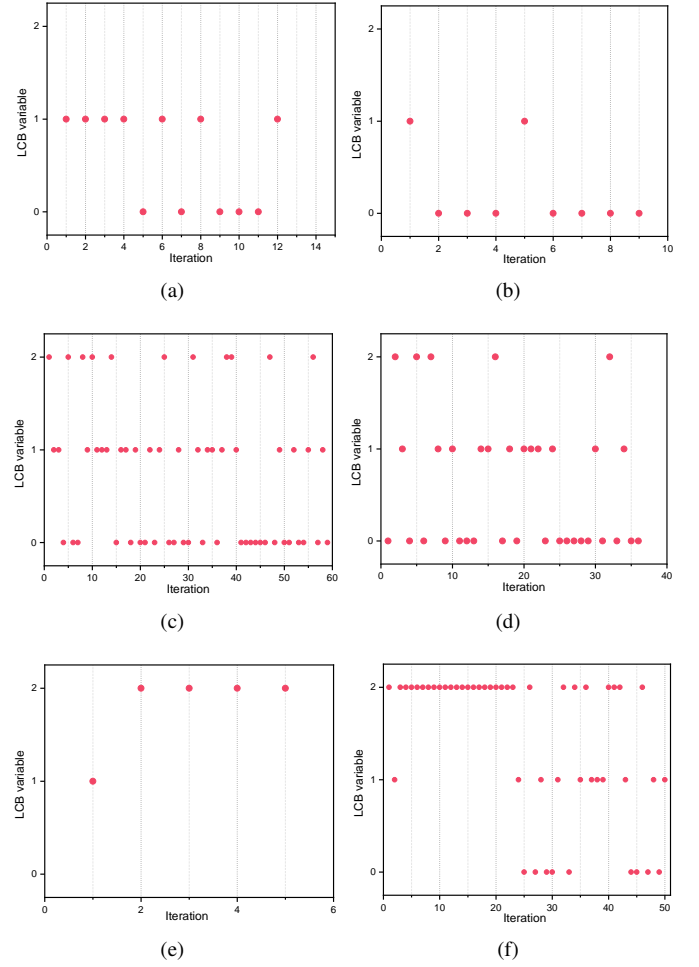


Fig. 15. The changes in the LCB variables corresponding to the HF EM simulation at each iteration. (a)-(b) Antenna 1; (c)-(d) Antenna 2; (e) Antenna 3; (f) Antenna 4.

time is reduced to 163 seconds. The target of Antenna 4 is to maximize the side-lobe level (SLL) within the input tolerance, which can be converted to a minimization problem

$$f(\mathbf{x}_i) = \min_{\mathbf{x}_i} \{-\{\tilde{R}_S(\mathbf{x}_i) + \omega_i R_S^s(\mathbf{x}_i)\}\}, i = 1, \dots, M. \quad (22)$$

First, the LF input parameters are randomly sampled by the LHS method around the design point, $N_L = 50$, and HF input data are selected from LF parameter vectors, $N_H = 20$. Moreover, the LF and HF data of the design point are added to the initial database. Fig. 15(f) shows the changes in LCB variables of the corresponding HF EM simulation in the WCP search process. The results of different search methods are shown in Fig. 14. Within 25 hours, the worst case SLL of the two optimization algorithms is approximately -16.26 dB, and MB-MLAO can reach this value in approximately 20.7 hours. Fig. 14(b) is the normalized pattern when the LCB constant equals 0, 1, 2, iterated 17, 19, 17 times, and the LCB constant equals [0, 1, 2], iterated 39 times. Since the input parameters corresponding to the worst cases of the three cases where the LCB constant is a fixed value are on the same searching boundary, the response curves are coincident. Within the specified number of 50 iterations, the worst value found by the proposed algorithm is -15.8 dB, and the optimization time

is 39.76 hours. However, for the algorithm with a fixed LCB constant, the worst value found within the specified number of iterations is not higher than -16 dB.

Fig. 15 illustrates that the LCB variable corresponding to the input parameters of the HF model verification is possibly different at different iterations. The proposed algorithm adaptively selects one LCB variable to obtain the optimal value under multiple branches for verification in the optimization process.

V. CONCLUSION

An MB-MLAO method that balances exploitation and exploration is proposed. The LCB variable is utilized to realize multibranch optimization, considering the most promising candidate points and sampling sparse areas. A variable-fidelity model reduces the computation time. The retraining and reprediction method improves the prediction accuracy and adaptively selects one parameter vector for verification by the HF EM model. To show the superiority of the MB-MLAO method, two test problems and four practical antennas are verified. For different antennas, the optimal LCB constant is usually a priori unknown. Even for the same antenna, due to different initial sample sets or different numbers of HF and LF data sets, the setting of the LCB constant affects the performance of the algorithm. For problems with different characteristics, different complexities, and different design specifications, the MB-MLAO method maintains effectiveness while obtaining robustness.

APPENDIX A

For ease of verification, the input parameters for four antennas before and after optimization are given in Tables II to VII.

TABLE II
ANTENNA 1, CASE I

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	48	21	9	12
l_0	18.23	19.65	19.50	19.19	19.62
w_0	12.43	12.00	12.00	12.00	12.00
a_0	0.39	0.30	0.30	0.30	0.43
l_p	13.37	14.00	13.26	13.10	13.55
w_p	6.30	6.00	5.97	5.88	5.99
s_0	1.19	1.20	1.20	1.20	1.20
Goal reached?	No	Yes	Yes	Yes	Yes

TABLE III
ANTENNA 1, CASE II

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	11	8	28	9
l_0	19.20	19.55	19.24	19.41	19.67
w_0	12.21	12.00	12.23	12.00	12.00
a_0	0.41	0.30	0.30	0.30	0.30
l_p	13.78	13.88	13.66	13.84	13.52
w_p	6.45	6.19	5.84	6.08	6.19
s_0	1.04	1.20	1.20	1.20	1.20
Goal reached?	Yes	Yes	Yes	Yes	Yes

TABLE IV
ANTENNA 2, CASE I

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	63	71	49	59
l_2	6.32	6.43	6.01	5.67	5.98
l_{s1}	22.63	23.48	23.54	24.33	22.79
l_{s2}	24.35	23.37	22.01	22.56	22.60
l_{s3}	6.60	7.33	5.41	5.93	5.98
w_{s1}	0.70	0.68	2.05	1.99	0.95
w_{s2}	2.24	1.95	2.47	2.21	2.42
l_p	33.75	34.11	33.76	32.64	34.12
w_p	32.93	32.90	32.92	32.85	33.49
l_{stub}	17.15	18.57	17.18	17.53	19.66
Goal reached?	Yes	No	No	No	Yes

TABLE VI
ANTENNA 3

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	14	9	18	5
l_1	2.79	2.42	2.57	2.40	2.40
l_2	0.72	0.78	0.71	0.90	0.70
l_3	1.81	1.94	1.84	2.20	1.80
l_4	12.13	12.2	12.15	12.25	12.07
l_{s1}	16.20	16.04	15.91	16.50	16.50
l_{s2}	0.65	0.78	0.55	0.65	0.80
w_3	1.12	1.28	1.17	1.30	1.30
w_{s1}	0.79	0.70	0.80	0.50	0.80
Goal reached?	Yes	Yes	Yes	Yes	Yes

TABLE V
ANTENNA 2, CASE II

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	71	43	44	36
l_2	6.32	6.43	5.94	6.79	5.84
l_{s1}	22.63	23.04	24.50	22.95	23.80
l_{s2}	24.35	23.98	23.53	22.30	22.97
l_{s3}	6.60	6.06	7.16	7.22	6.68
w_{s1}	0.70	1.59	1.28	1.03	1.31
w_{s2}	2.24	2.12	2.38	2.13	2.40
l_p	33.75	33.94	33.01	34.93	33.45
w_p	32.93	32.70	33.17	33.98	33.73
l_{stub}	17.15	17.81	18.95	18.56	18.59
Goal reached?	No	No	No	No	Yes

REFERENCES

- [1] B. Zhang and Y. Rahmat-Samii, “Robust optimization with worst case sensitivity analysis applied to array synthesis and antenna designs,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 1, pp. 160–171, 2018.
- [2] J. A. Easum, J. Nagar, P. L. Werner, and D. H. Werner, “Efficient multiobjective antenna optimization with tolerance analysis through the use of surrogate models,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 12, pp. 6706–6715, 2018.
- [3] Q. Wu, W. Chen, C. Yu, H. Wang, and W. Hong, “Multilayer machine learning-assisted optimization-based robust design and its applications to antennas and arrays,” *IEEE Trans. Antennas Propag.*, pp. 1–6, 2021.
- [4] S. D. Rogers, C. M. Butler, and A. Q. Martin, “Design and realization of GA-optimized wire monopole and matching network with 20:1 bandwidth,” *IEEE Trans. Antennas Propag.*, vol. 51, no. 3, pp. 493–502, Mar. 2003.
- [5] Q. Wu, J. Hirokawa, J. Yin, C. Yu, H. Wang, and W. Hong, “Millimeter-wave multibeam endfire dual-circularly polarized antenna array for 5G wireless applications,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 9, pp. 4930–4935, Sep. 2018.
- [6] J. Robinson and Y. Rahmat-Samii, “Particle swarm optimization in electromagnetics,” *IEEE Trans. Antennas Propag.*, vol. 52, no. 2, pp. 397–407, Feb. 2004.
- [7] J. Branke and C. Schmidt, “Faster convergence by means of fitness estimation,” *Soft Comput.*, vol. 9, no. 1, pp. 13–20, 2005.
- [8] L.-Y. Xiao, W. Shao, F.-L. Jin, and B.-Z. Wang, “Multiparameter modeling with ANN for antenna design,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 7, pp. 3718–3723, Apr. 2018.
- [9] L. Cui, Y. Zhang, R. Zhang, and Q. H. Liu, “A Modified Efficient KNN Method for Antenna Optimization and Design,” *IEEE Trans. Antennas Propag.*, vol. 68, no. 10, pp. 6858–6866, Oct. 2020.
- [10] D. R. Prado, J. A. López-Fernández, G. Barquero, M. Arrebola, and F. Las-Heras, “Fast and accurate modeling of dual-polarized reflectarray unit cells using support vector machines,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 3, pp. 1258–1270, Mar. 2018.
- [11] S. Koziel, S. Ogurtsov, I. Couckuyt, and T. Dhaene, “Variable-fidelity electromagnetic simulations and co-kriging for accurate modeling of antennas,” *IEEE Trans. Antennas Propag.*, vol. 61, no. 3, pp. 1301–1308, 2013.
- [12] Q. Wu, H. Wang, and W. Hong, “Multistage collaborative machine learning and its application to antenna modeling and optimization,” *IEEE Trans. Antennas Propag.*, vol. 68, no. 5, pp. 3397–3409, May 2020.
- [13] Z. Zhang, H. C. Chen, and Q. S. Cheng, “Surrogate-assisted quasi-newton enhanced global optimization of antennas based on a heuristic hypersphere sampling,” *IEEE Trans. Antennas Propag.*, vol. 69, no. 5, pp. 2993–2998, 2021.
- [14] B. Liu, D. Zhao, P. Reynaert, and G. Gielen, “Synthesis of integrated passive components for high-frequency RF ICs based on evolutionary computation and machine learning techniques,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 10, pp. 1458–1468, Oct. 2011.
- [15] B. Liu, Q. Zhang, and G. Gielen, “A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems,” *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 180–192, Apr. 2014.
- [16] M. T. Emmerich, K. C. Giannakoglou, and B. Naujoks, “Single- and multiobjective evolutionary optimization assisted by Gaussian random field metamodels,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 421–439, Aug. 2006.
- [17] 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, 1998.
- [18] Q. Zhang, W. Liu, E. Tsang, and B. Virginas, “Expensive multiobjective optimization by MOEA/D with Gaussian process model,” *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 456–474, Jun. 2010.

TABLE VII
ANTENNA 4

	Initial	$\omega = 0$	$\omega = 1$	$\omega = 2$	$\omega = [0, 1, 2]$
Iter.	-	17	19	17	39
w_1	0.56	0.66	0.66	0.66	0.66
w_2	0.70	0.60	0.60	0.60	0.80
w_3	0.84	0.94	0.94	0.94	0.74
w_4	1.12	1.32	1.32	1.32	1.32
w_5	1.19	0.99	0.99	0.99	0.99
w_6	1.40	1.60	1.60	1.60	1.20
w_7	1.47	1.67	1.67	1.67	1.27
w_8	1.47	1.27	1.27	1.27	1.67
w_9	1.47	1.67	1.67	1.67	1.67
w_{10}	0.84	0.94	0.94	0.94	0.94
w_{11}	0.70	0.60	0.60	0.60	0.80
w_{12}	0.21	0.26	0.26	0.26	0.16
Worst SLL (dB)		-16.26	-16.26	-16.26	-15.80

- [19] M. Emmerich, A. Giotis, M. Özdemir, T. Bäck, and K. Giannakoglou, “Metamodel—assisted evolution strategies,” in *Parallel Problem Solving from Nature — PPSN VII*. Springer Berlin Heidelberg, 2002, pp. 361–370.
- [20] M. W. Trosset and V. Torczon, “Numerical optimization using computer experiments,” Institute for Computer Applications in Science and Engineering, Hampton Va, Tech. Rep., 1997.
- [21] M. C. Kennedy and A. O’Hagan, “Predicting the output from a complex computer code when fast approximations are available,” *Biometrika*, vol. 87, no. 1, pp. 1–13, 2000.
- [22] P. Perdikaris, M. Raissi, A. Damianou, N. D. Lawrence, and G. E. Karniadakis, “Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling,” *Proc. R. Soc. A.*, vol. 473, no. 2198, 2017.
- [23] M. Stein, “Large sample properties of simulations using latin hypercube sampling,” *Technometrics*, vol. 29, no. 2, pp. 143–151, 1987.
- [24] A. Paleyes, M. Pullin, M. Mahsereci, N. Lawrence, and J. Gonzalez, “Emulation of physical processes with emukit,” in *Second Workshop on Machine Learning and the Physical Sciences, NeurIPS*, 2019.
- [25] Jazzbin and et.al., “geatpy: The genetic and evolutionary algorithm toolbox with high performance in python,” 2020.
- [26] B. Liu, S. Koziel, and Q. Zhang, “A multi-fidelity surrogate-model-assisted evolutionary algorithm for computationally expensive optimization problems,” *J. Comput. Sci.*, vol. 12, pp. 28–37, 2016.
- [27] T. G. Spence and D. H. Werner, “A novel miniature broadband/multiband antenna based on an end-loaded planar open-sleeve dipole,” *IEEE Trans. Antennas Propag.*, vol. 54, no. 12, pp. 3614–3620, Dec. 2006.
- [28] J.-F. Qian, F.-C. Chen, and Q.-X. Chu, “A novel tri-band patch antenna with broadside radiation and its application to filtering antenna,” *IEEE Trans. Antennas Propag.*, vol. 66, no. 10, pp. 5580–5585, 2018.
- [29] CST Microwave Studio. Ver. 2020, Dassault Systèmes, Darmstadt, Germany.
- [30] J. Xie, J. Yin, Q. Wu, and H. Wang, “Low-sidelobe series-fed microstrip antenna array for 77 GHz automotive radar applications,” in *UK-Europe-China Workshop Millim. Waves Terahertz Technol., UCMMT - Proc.*, vol. 1, 2018, pp. 1–3.