Global Optimization of Microwave Filters Based on a Surrogate Model-Assisted Evolutionary Algorithm

Bo Liu, *Member, IEEE*, Hao Yang, and Michael J. Lancaster, *Senior Member, IEEE*

***Abstract*—Local optimization is a routine approach for full-wave optimization of microwave filters. For filter optimization problems with numerous local optima or where the initial design is not near to the optimal region, the success rate of the routine method may not be high. Traditional global optimization techniques have a high success rate for such problems, but are often prohibitively computationally expensive considering the cost of full-wave electromagnetic simulations. To address the above challenge, a new method, called surrogate model-assisted evolutionary algorithm for filter optimization (SMEAFO), is proposed. In SMEAFO, considering the characteristics of filter design landscapes, Gaussian process surrogate modeling, differential evolution operators, and Gaussian local search are organized in a particular way to balance the exploration ability and the surrogate model quality, so as to obtain high-quality results in an efficient manner. The performance of SMEAFO is demonstrated by two real-world design cases (a waveguide filter and a microstrip filter), which do not appear to be solvable by popular local optimization techniques. Experiments show that SMEAFO obtains high-quality designs comparable with global optimization techniques but within a reasonable amount of time. Moreover, SMEAFO is not restricted by certain types of filters or responses. The SMEAFO-based filter design optimization tool can be downloaded from http://fde.cadescenter.com.**

***Index Terms*—Design optimization, design tools, evolutionary computation, Gaussian process (GP), metamodeling, microwave filters.**

I. Introduction

Microwave filter design can be formulated as an optimization problem. Among various optimization methods, evolutionary algorithms (EAs) are being widely used for microwave design optimization due to their high global optimization ability, free of a good initial design, wide applicability, and robustness [1]–[3]. Moreover, they are embedded in most commercial electromagnetic (EM) simulation tools, such as CST Microwave Studio. However, EAs are seldom applied to microwave filter design, because full-wave EM simulations are often needed to obtain accurate performance evaluation, which are computationally expensive. Considering thousands to tens of thousands of EM simulations needed for a standard EA to get the optimum, the filter design optimization time can be unbearable (e.g., several months).

To obtain an optimal design in a reasonable timeframe, local optimization from an initial design has become a routine approach for filter design optimization during the last decade. Derivative-based local optimization methods (e.g., sequential quadratic programming [4]) and derivative free local optimization methods (e.g., the Nelder–Mead simplex method [5]) are widely applied. Because the quality of the initial design is essential for the success of local optimization, a lot of research has been done aiming to find a reasonably good initial design efficiently. Available methods mainly include employing equivalent circuit [6], low-fidelity EM model [7] for a preliminary relatively low-cost optimization, and coupling matrix fitting [8].

To further improve the efficiency of local optimization, the space mapping technique [9] is widely used. Several important improvements have been made to enhance the reliability and the efficiency of traditional space mapping, such as introducing the human design intuition [10], altering an EM model by embedding suitable tuning elements (port tuning) [11], and the multilevel method [12]. The port tuning method has shown great success in commercial applications for planar filters. Methods based on integrating human design intuition and port tuning have obtained optimal designs for some filters whose initial designs are not near the optimal region. Adjoint sensitivity is also introduced to replace traditional gradient-based local optimization techniques, and shows great speed improvement [13].

Although many filters have been successfully designed using the available techniques, and some of them even only need a few high-fidelity EM simulations, available methods still face severe challenges when the initial design is not near the optimal region and/or the filter design landscape has many local optima (not smooth enough) [10]. Unfortunately, this happens to many microwave filter design problems, and this problem is the target of this paper. Clearly, traditional space mapping and adjoint sensitivity techniques are difficult to provide a generic solution to this issue, because their main goal is to improve the efficiency of local optimization rather than improve the optimization capacity (i.e., jumping out of

Manuscript received July 5, 2016; revised October 25, 2016; accepted December 24, 2016. Date of publication March 2, 2017; date of current version June 2, 2017. This work was supported by the U.K. Engineering and Physical Science Research Council under Project EP/M016269/1.

B. Liu is with the School of Electrical, Electronic, and System Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, U.K., and also with the Department of Computing, Wrexham Glyndwr University, Wrexham LL11 2AW, U.K. (e-mail: b.liu.3@bham.ac.uk; liubo168@gmail.com).

H. Yang and M. J. Lancaster are with the School of Electrical, Electronic, and System Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, U.K. (e-mail: hxy297@student.bham.ac.uk; m.j.lancaster@bham.ac.uk).

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

Digital Object Identifier 10.1109/TMTT.2017.2661739

local optima). In recent years, some novel methods have been proposed to improve the optimization capacity while keeping the efficiency improvement, but they often concentrate on a certain type of filter or response [6].

Also, with the rapid improvement of computing power and numerical analysis techniques, high-fidelity EM simulation of many microwave filters can be completed within 20 min. Although directly employing EAs is still prohibitively computationally expensive, developing *widely applicable* methods with *largely improved optimization ability* compared with local optimization, but using *a practical timeframe* (e.g., within several days) for the targeted problem is of great importance to complement the state of the arts.

An alternative is surrogate model-assisted EAs (SAEAs), which introduce surrogate modeling to EAs. In the context of filter optimization, a surrogate model is a computationally cheap mathematical model approximating the output of EM simulations, which is often constructed by statistical learning techniques and is widely used in space mapping. By coupling surrogate models with an EA, some of the EM simulations can be replaced by the surrogate model predictions, and the computational cost can, therefore, be reduced significantly. SAEA is attracting increasing attention in the computational intelligence field, and various new SAEAs have been proposed.

However, most of the available SAEAs are not suitable for filter optimization. Besides the efficiency issue, microwave filter optimization has two difficulties:

1) A filter is a narrowband device, and the optimal region is often very narrow.
2) There often exist numerous local optima in the landscape, especially for high-order filters.

Because of different tradeoffs between the exploration ability and the surrogate model quality, most available SAEAs can get either the optimal solution, but need more than necessary EM simulations causing very long optimization time or spend reasonable time but miss the optimal solution. The reasons will be described in Section III-A.

To address this challenge, a new method is proposed, called SAEA for filter optimization (SMEAFO). The main innovation of SMEAFO is the new SAEA framework balancing the exploration ability and the surrogate model quality considering the characteristics of filter design landscape. SMEAFO targets at filter optimization problems with numerous local optima and/or where the initial design is far from the optimal region, aiming to the following:

1) achieve comparable results with standard EAs (often have very high success rate and are considered as the best in terms of solution quality);
2) obtain significant speed improvement compared with standard EAs and complete the optimization in a reasonable timeframe (several hours to several days) for problems with less than 20 min per EM simulation;
3) general enough for most kinds of filters without considering specific properties of the targeted filter.

The remainder of this paper is organized as follows. Section II introduces the basic techniques. Section III introduces the SMEAFO algorithm, including its main ideas, design of each algorithmic component, its general framework, and parameter settings. Section IV presents a waveguide filter and a microstrip filter that do not appear to be solvable by available popular local optimization techniques to show the performance of SMEAFO. Comparisons with the standard differential evolution (DE) algorithm are also provided. The concluding remarks are presented in Section V.

II. Basic Techniques

A. Gaussian Process Surrogate Modeling

Among various surrogate modeling methods, Gaussian process (GP) machine learning [14] is selected for SMEAFO. The main reason is that the prediction uncertainty of GP has a sound mathematical background, which is able to take advantage of prescreening methods [15] for surrogate model-based optimization. A brief introduction is as follows. More details are in [14].

Given a set of observations $x = (x^1, \ldots, x^n)$ and $y = (y^1, \ldots, y^n)$, GP predicts a function value $y(x)$ at some design point x by modeling $y(x)$ as a Gaussian distributed stochastic variable with mean μ and variance σ^2. If the function is continuous, the function values of two points x^i and x^j should be close if they are highly correlated. In this paper, we use the Gaussian correlation function to describe the correlation between two variables

$$\mathrm{corr}(x^i, x^j) = \exp\left(-\sum_{l=1}^{d} \theta_l |x_l^i - x_l^j|^2\right) \tag{1}$$

where d is the dimension of x and θ_l is the correlation parameter, which determines how fast the correlation decreases when x^i moves in the l-direction. The values of μ, σ, and θ are determined by maximizing the likelihood function that $y = y^i$ at $x = x^i (i = 1, \ldots, n)$. The optimal values of μ and σ can be found by setting the derivatives of the likelihood function to 0 and solve the equations, which are as follows:

$$\hat{\mu} = (I^T R^{-1} y)^{-1} I^T R^{-1} y \tag{2}$$

$$\hat{\sigma}^2 = (y - I\hat{\mu})^T R^{-1} (y - I\hat{\mu}) n^{-1} \tag{3}$$

where I is an $n \times 1$ vector with all elements having the value of one and R is the correlation matrix

$$R_{i,j} = \mathrm{corr}(x^i, x^j), \quad i, j = 1, 2, \ldots, n. \tag{4}$$

Using the GP model, the function value $y(x^*)$ at a new point x^* can be predicted as (x^* should be included in the correlation matrix)

$$\hat{y}(x^*) = \hat{\mu} + r^T R^{-1} (y - I\hat{\mu}) \tag{5}$$

where

$$r = [\mathrm{corr}(x^*, x^1), \mathrm{corr}(x^*, x^2), \ldots, \mathrm{corr}(x^*, x^n)]^T. \tag{6}$$

The measurement of the uncertainty of the prediction (mean square error), which is used to access the model accuracy, can be described as

$$\hat{s}^2(x^*) = \hat{\sigma}^2[I - r^T R^{-1} r + (I - r^T R^{-1} r)^2 (I^T R^{-1} I)^{-1}]. \tag{7}$$

To make use of the prediction uncertainty to assist SAEA, the lower confidence bound prescreening [15], [16] is selected. We consider the minimization of $y(x)$ in this paper. Given the predictive distribution $N(\hat{y}(x), s^2(x))$ for $y(x)$, a lower confidence bound prescreening of $y(x)$ can be defined as [16]

$$y_{lcb}(x) = \hat{y}(x) - \omega s(x) \\ \omega \in [0, 3] \tag{8}$$

where ω is a constant, which is often set to two to balance the exploration and exploitation ability [15].

In this paper, we use the ooDACE toolbox [17] to implement the GP surrogate model.

B. Differential Evolution

In SMEAFO, the DE algorithm [18] is selected as the global search engine. DE outperforms many EAs for continuous optimization problems [18] and also shows advantages for EM design optimization problems among various EAs [19]. DE is an iterative method. In each iteration, the mutation operator is firstly applied to generate a population of mutant vectors. A crossover operator is then applied to the mutant vectors to generate a new population. Finally, selection takes place and the corresponding candidate solutions from the old population and the new population compete to comprise the population for the next iteration.

In DE, mutation is the main approach to explore the design space. There are a few different DE mutation strategies trading off the convergence speed and the population diversity (implying higher global exploration ability) in different manners. Arguably, the three DE mutation strategies that are widely used in engineering design optimization are as follows.

1) *Mutation Strategy 1*: DE/best/1

$$v^i = x^{\text{best}} + F \cdot (x^{r_1} - x^{r_2}) \tag{9}$$

where x^{best} is the best individual in P (the current population) and x^{r_1} and x^{r_2} are two different solutions randomly selected from P and are also different from x^{best}. v^i is the ith mutant vector in the population after mutation. $F \in (0, 2]$ is a control parameter, often called the scaling factor [18].

2) *Mutation Strategy 2*: DE/rand/1

$$v^i = x^{r_3} + F \cdot (x^{r_1} - x^{r_2}). \tag{10}$$

Compared with DE/best/1, x^{best} is replaced by a randomly selected solution x^{r_3} that is also different from x^{r_1} and x^{r_2}.

3) *Mutation Strategy 3*: DE/current-to-best/1[1]

$$v^i = x^i + F \cdot (x^{\text{best}} - x^i) + F \cdot (x^{r_1} - x^{r_2}) \tag{11}$$

where x^i is the ith vector in the current population.

Crossover is then applied to the population of mutant vectors to produce the child population U, which works as follows.

1) Randomly select a variable index $j_{\text{rand}} \in \{1, \ldots, d\}$.

2) For each $j = 1$ to d, generate a uniformly distributed random number rand from (0, 1) and set

$$u_j^i = \begin{cases} v_j^i, & \text{if (rand} \leq \text{CR)} | j = j_{\text{rand}} \\ x_j^i, & \text{otherwise} \end{cases} \tag{12}$$

where CR $\in [0, 1]$ is a constant called the crossover rate.

Following that, the selection operation decides on the population of the next iteration, which is often based on a one-to-one greedy selection between P and U.

C. Gaussian Local Search

Gaussian local search is a verified effective method for elaborate search in a local area [20]. Gaussian local search is often used for enhancing local search ability of EAs. In SMEAFO, the following implementation is used:

$$xg_j^i = \begin{cases} x_j^i + N(0, \sigma_j^{\text{gls}}), & \text{if rand} \leq \frac{1}{d} \\ x_j^i, & \text{otherwise;} \\ & j = 1, 2, \ldots, d \end{cases} \tag{13}$$

where $N(0, \sigma_j^{\text{gls}})$ is a Gaussian distributed random number with a standard deviation of σ_j^{gls} and rand is a uniformly distributed random number from (0, 1).

III. SMEAFO Algorithm

A. Challenges and Main Ideas of SMEAFO

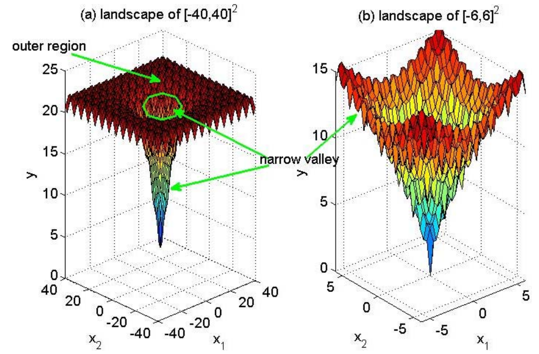


Fig. 1. Illustrative figure of filter design landscape: the Ackley function [21] is used.

The SMEAFO algorithm is an SAEA. Integrating a surrogate model into global optimization is much more difficult than integrating it into space mapping because there is no information of the optimal region. Recall the two major difficulties for filter optimization described in Section I (an illustrative figure is Fig. 1): 1) the optimal region is located in a (very) narrow valley of the design space and 2) there are often numerous local optima. The SAEA, therefore, should have sufficient exploration ability to jump out of local optima in the outer region so as to find the narrow valley and to jump out of local optima within it. Although this is often achievable for a modern standard EA, SAEAs may not have the same exploration ability due to the surrogate model prediction uncertainty, i.e., some optimal designs may be predicted wrong and then the SAEA search is guided to wrong directions.

[1] This mutation strategy is also referred to as DE/target-to-best/1.

High exploration ability indicates getting access to diverse candidate designs. To make a good prediction of them, more training data points through EM simulations are necessary to maintain the surrogate model quality, which decreases the efficiency. Finding an appropriate balance between the exploration ability and the efficiency for filter design landscape is the main challenge of SMEAFO.

The required exploration ability in the filter optimization process is different from time to time. Instead of using a fixed SAEA with a certain exploration ability, it is natural to divide it into the *exploration phase* aiming to find a near optimal region and the *exploitation phase* aiming to obtain the final optimal design from near-optimal designs. The latter phase requires less exploration ability (indicating more space for efficiency) without sacrificing the solution quality. Various methods can be used for the exploitation phase, and space mapping is compatible. Because space mapping is sometimes sensitive to the surrogate model type and settings [7], a surrogate model-assisted Gaussian local search method is used in SMEAFO for the sake of generality. Now the major challenge is the exploration phase providing both sufficient exploration ability and efficiency.

To balance the exploration ability and the efficiency, available SAEAs can be mainly classified into "conservative" SAEAs and "active" SAEAs. Conservative SAEAs [22], [23] emphasize the exploration ability. These methods begin with a standard EA for certain iterations aiming to collect training data points that are able to build a reasonably good global surrogate model and then iteratively improve the solution quality and the surrogate model quality in the consecutive search. Thus, the exploration ability can benefit a lot at the cost of a considerable computing overhead for standard EA-based iterations. When applied to filter optimization, much of this computing overhead is wasted because they are collecting training data points for modeling the outer region instead of the narrow valley where the optimal design is located.

Active SAEAs, in contrast, emphasize the efficiency [15]. These methods perform exact expensive evaluations to the "optimal" solutions predicted by the existing surrogate model, despite that its quality may not be good enough. The number of expensive exact evaluations is therefore highly reduced, but the exploration ability becomes a weakness. Prescreening methods [16], [24] are used to assist jumping out of local optima, but they cannot fully solve the problem. In [15], tests on the Ackley benchmark problem (with a narrow valley and many local optima) [21] (Fig. 1) show that such SAEAs are not able to jump out of local optima. Hence, the exploration ability is insufficient for filter optimization.

The exploration phase of SMEAFO follows the idea of active SAEAs to avoid consuming considerable EM simulations to nonoptimal regions. To largely improve the exploration ability compared with existing active SAEAs, two questions are focused: 1) what is the search method to obtain sufficient exploration ability? and 2) how to build surrogate models of sufficient quality using as few samples as possible (for the sake of efficiency) in order to support the exploration ability? This is achieved by the combination of a novel surrogate model assisted search method with specific DE mutation and training sample selection methods, which will be detailed in Section III-B.

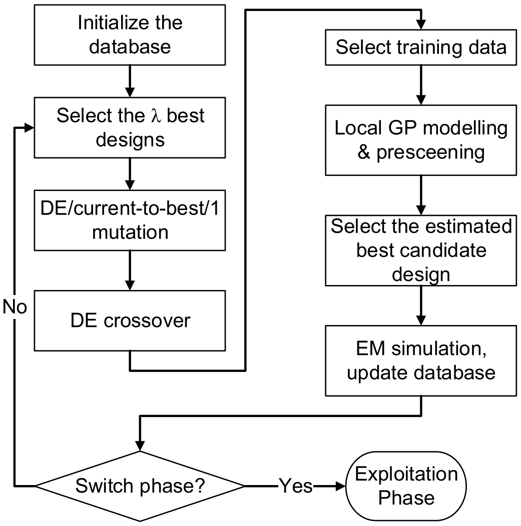

Initialize the database
Select the λ best designs
DE/current-to-best/1 mutation
DE crossover
No
Select training data
Local GP modelling & presceening
Select the estimated best candidate design
EM simulation, update database
Switch phase?
Yes
Exploitation Phase

Fig. 2. Flow diagram of the exploration phase.

B. Design of the Exploration Phase

The general framework of the exploration phase is shown in Fig. 2, which consists of the following steps.

Step 1: Sample λ (often small) candidate designs, perform EM simulations of all of them, and let them form the initial database.

Step 2: Select the λ best designs from the database based on the objective function values to form a population P.

Step 3: Apply the DE/current-to-best/1 mutation (11) and the crossover operator (12) on P to generate λ child solutions.

Step 4: For each child solution, select training data points and construct a local GP surrogate model.

Step 5: Prescreen the λ child solutions generated in Step 3 using the lower confidence bound method. Estimate the best child solution based on the lower confidence bound values.

Step 6: Perform EM simulation to the estimated best child solution from Step 5. Add this design and its performance (EM simulation result) to the database. Go back to Step 2 until switching to the exploitation phase.

A main difference compared with available active SAEAs is that a standard EA process is not adopted; instead, only the predicted best candidate design is simulated and the current best λ candidate designs are used as the new population in each iteration. This new SAEA framework improves the locations of training data points. It is well known that the number of training data points affects the quality of the surrogate model, while their locations are often overlooked. With the same number of training data points, it is intuitive that using training data points located near the points waiting to be predicted (child population in Step 3) can obtain surrogate model(s) with better quality. This is implemented in Steps 2–6.

From Step 2 to Step 6, in each iteration, the λ current best candidate solutions construct the parent population (it is reasonable to assume that the search focuses on the promising region) and the best candidate design based on prescreening in the child population is selected to replace the worst one in the parent population. Hence, only at most one candidate is changed in the parent population in each iteration, so the best candidate in the child solutions in several consecutive iterations may not be far from each other (they will then be simulated and are used as training data points). Therefore, the training data points describing the current promising region can be much denser compared with those generated by a standard EA population updating mechanism, which may spread in different regions of the design space, while there may not be sufficient training data points around the candidate solutions to be prescreened.

Using the database with improved sample locations for surrogate modeling, a consecutive critical problem is selecting samples from it which will be used as the training data points. Most SAEAs build a single surrogate model for predicting the child population. For example, a certain number of evaluated promising solutions (ranked by fitness function values) [15] or latest solutions are used to build a model for the child population. But such methods are not suitable for the targeted problem because of the two design landscape characteristics of microwave filters (Section III-A). In particular, due to the narrow valley where the optimal design is located, a promising point that is located near it may be predicted to be not promising when many training data points are in the outer nonoptimal region. The reason is that the hyperparameters in (1) are highly likely to be poorly estimated in likelihood function optimization when the number of training data points near it is insufficient. Therefore, in SMEAFO, a local GP surrogate model is built for each child solution using τ nearest samples (based on Euclidean distance). This means that λ separate local GP models are built in each iteration.

With improved surrogate model quality, appropriate search operators should be selected to provide neither insufficient nor excessive population diversity, which directly determine the exploration ability. Intuitively, DE/best/1 (9) may not have sufficient population diversity, because the added diversity into the current best design is not large. Note that although there exist SAEAs with DE/best/1 showing success [25], the optima of the test problems are not located in a narrow valley. In contrast, DE/rand/1 (10) may introduce too much population diversity. DE/current-to-best/1 (11) is in the middle. Pilot experiments on the Ackley benchmark problem [21] are carried out. Results show that DE/current-to-best/1 just gets an appropriate balance of the population diversity and the surrogate model quality (almost 100% getting very near to the global optimum) with the new GP model-assisted search framework, while DE/rand/1 performs the worst because excessive diversity suffers the surrogate model quality.

It has to be noted that the above particular surrogate model-assisted search method, the training data selection method for GP modeling (building a separate GP model for each child solution), and the above DE mutation operator (DE/current-to-best/1) must be used together. Pilot experiments on real-world filters show that when any of the factors is altered, the algorithm often fails to find the narrow valley or the performance becomes unstable.

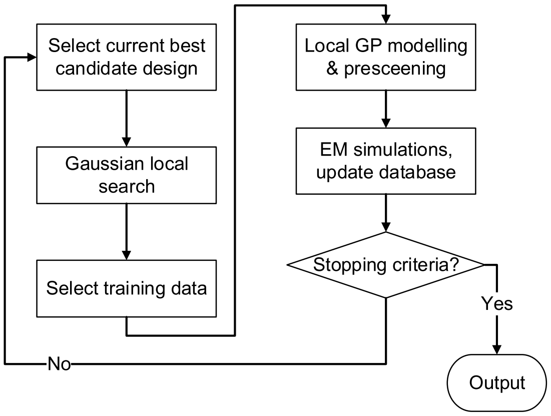

Select current best candidate design
Gaussian local search
Select training data
Local GP modelling & presceening
EM simulations, update database
Stopping criteria?
Yes
No
Output

Fig. 3. Flow diagram of the exploitation phase.

Note that the lower confidence bound method also contributes to the algorithm performance. In Step 5, instead of the predicted value of the GP model, the lower confidence bound value (8) is used for ranking. The use of lower confidence bound can balance the search between present promising regions [i.e., with low $\hat{y}(x)$ values in (8)] and less explored regions [i.e., with high $s(x)$ values], so as to improve the ability of an SAEA to jump out of local optima. Reference [15] provides more details.

For initial population generation (Step 1), each candidate solution is calculated by (14) if an initial design is available; otherwise, it is generated randomly within the design space

$$x_j = x_j^{\text{init}} + N(0, \sigma_{\text{init}}), \quad j = 1, 2, \ldots, d \tag{14}$$

where x^{init} is the initial design and σ_{init} is the standard deviation of the added Gaussian distributed random number. The value of σ_{init} is roughly estimated according to the response of the initial design. Pilot experiments show that the initial population is far from optimal no matter if using (14) or random generation due to the landscape characteristics of microwave filters and the quality of the initial designs. Note that SMEAFO performs global optimization and a poor initial population is not a problem and is even assumed.

C. Design of the Exploitation Phase

The general framework of the exploitation phase is shown in Fig. 3, which consists of the following steps.

Step 1: Perform Gaussian local search from the current best design in the database to generate d (number of design variables) solutions.

Step 2: For each solution from Step 1, select training data points using the method in Section III-B and construct a local GP surrogate model.

Step 3: Prescreen the d solutions generated in Step 1 using the lower confidence bound method. For each of them, if the lower confidence bound value is better than the current best design, perform an

EM simulation to it. Add this design and its performance (the EM simulation result) to the database.

Step 4: If a preset stopping criterion (e.g., computing budget) is met, output the best solution from the database; otherwise, go back to Step 1.

The goal of the exploitation phase is to obtain the final optimal design from a near optimal design based on a surrogate-based local search method with largely reduced exploration ability. Although heuristic local search methods themselves are not complex, a common challenge is the adaptation of critical parameters, including the starting condition and the scale of exploitation [26].

Note that there is no clear threshold to divide exploration and exploitation in a search process because "near optimal" is an empirical definition [26]. However, an appropriate definition of the starting condition of the exploitation phase is important for SMEAFO. Early starting of this phase may make the algorithm trapped in a local optimum, while late starting decreases the efficiency. In SMEAFO, we use the average standard deviation of the current population P, σ_P, to reflect the population diversity or progress of SMEAFO. Often, the value of σ_P first increases (exploring the design space) and then decreases (converge to the optimal area) in an SMEAFO run. We set 10% of the maximum σ_P in the exploration phase as the threshold to start the exploitation phase.

For the sake of generality, a verified effective method for elaborate search, Gaussian local search (Section II-C), is used in this phase. σ_{gls} is a critical parameter of Gaussian local search and is problem dependent. However, with the help of the exploration phase, it is set self-adaptively as

$$\sigma_j^{\text{gls}} = 0.5 \times \text{std}(\text{PB}_j), \quad j = 1, 2, \ldots, d \tag{15}$$

where PB is the best d candidate designs in the database and std is the standard deviation. This indicates that 95.4% (2σ value) of the candidate designs generated by Gaussian local search are within the standard deviation of the best d available candidate designs, which are already in a small region. This is in line with the basic idea of this phase (performing local exploitation around the current best design). With the update of PB, the σ^{gls} is self-adapted. Experiments on mathematical benchmark problems and eight real-world filter design problems verified empirical settings of σ_P and the self-adaptive setting of σ^{gls}.

Considering the surrogate model quality, because this phase performs local search, the database provided by the exploration phase is a good starting pool of training data points. The training data pool is also updated adaptively by Step 3 supporting the consecutive local search. The lower confidence bound value is used in Step 3 to avoid missing potentially optimal solutions, which also provides more samples around the optimal region.

D. Parameter Settings

Besides the self-adaptive parameters and the threshold value to start the exploitation phase (they are no longer parameters), remaining parameters are the DE parameters (λ, F, and CR), the number of training data points (τ) for each solution waiting to be prescreened (Step 4 in the exploration phase and Step 2 in the exploitation phase), and σ_{init} (Step 1 in the exploration phase).

The DE parameters have clear setting rules. Following [18], we suggest $F = 0.8$, $\text{CR} = 0.8$, and $\lambda = 50$. We suggest $\tau = 8 \times d$. This is based on the empirical rule in [24] and [25] for online surrogate modeling, and pilot experiments show a success. Note that in all the test problems, the same set of the above parameters is used. σ_{init} is a rough estimation of the scale to be added to the initial design if it exists. If the response of the initial design is far from anticipated, a larger σ_{init} can be used; otherwise, a smaller one may be used. This parameter is not sensitive for most filter design cases because no optimal solution is expected in the initial population. For some very challenging cases, using a smaller value is recommended to prevent the valley from becoming too narrow, so as to improve the solution quality and efficiency. The use of (14) is because using information from the initial design (although may have low quality) is better than random initialization.

IV. Experimental Results and Comparisons

SMEAFO has been tested by eight real-world filter design problems (five waveguide filters, one hairpin filter, one microstrip filter, and one diplexer). The initial designs are obtained by equivalent circuits or coupling matrix fitting [27]. The number of design variables varies from 5 to 22. The number of orders varies from 3 to 16. SMEAFO obtains high-quality results to all of them taking from 10 h to four days. We have not successfully solved six out of eight problems by popular local optimization-based methods.

In this section, two examples are used to demonstrate SMEAFO for different kinds of filter optimization problems with different challenges. The first one is a waveguide filter, and the initial design is obtained by coupling matrix fitting. Unfortunately, the initial response is far from the design specifications. The second one is a microstrip filter. The initial design is obtained by an equivalent circuit optimization, and the initial response is reasonably good. However, this seemingly easy problem is, in fact, difficult because the design landscape is very rugged, making local search methods fail to jump out of local optima in the narrow valley.

For the first example, ten runs of SMEAFO with independent random numbers (including initialization) are carried out to test the robustness of SMEAFO and the results are analyzed statistically. A comparison with standard DE is also carried out. Because the advantages of the DE algorithm compared with some other popular EAs (e.g., genetic algorithm and particle swarm optimization) in microwave engineering have been demonstrated in [28], such comparisons will not be repeated here. For the second example, only a single run of SMEAFO is carried out because standard DE is not affordable in terms of the computing overhead. The ability to handle larger search space is especially interesting for filter optimization because this is a major challenge of filter landscapes (Section I). This example has 12 design variables, which is relatively large for filter optimization, and we further intentionally expand the search ranges of each design variable

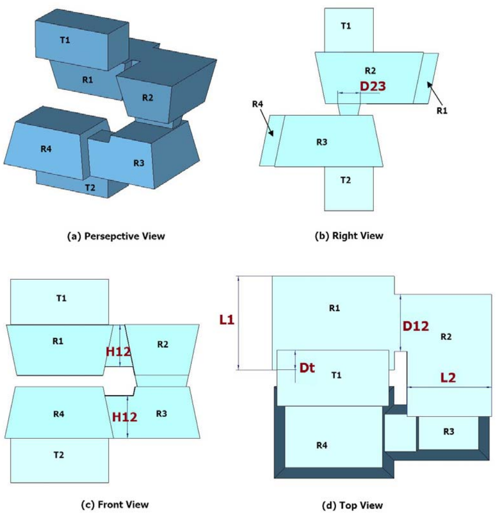


Fig. 4. Waveguide filter: T1, input test port, T2: output test port, and R1–R4: resonators.

TABLE I

RANGES OF THE SIX DESIGN VARIABLES (ALL SIZES IN MILLIMETERS) FOR EXAMPLE 1

$Variables$	$D12$	$D23$	Dt	$H12$	$L1$	$L2$
Lower bound	0.2	0.1	0.05	0.05	0.5	0.5
Upper bound	0.6	0.3	0.3	0.43	0.9	0.9

to make the optimal valley even narrower, so as to verify the capability of SMEAFO on an extreme case.

Both examples are constrained optimization problems. The penalty function method [29] is used to handle the constraints, and the penalty coefficient is set to 50. The examples are run on a PC with Intel 3.5-GHz Core i7 CPU and 8-GB RAM under Windows operating system. CST is used as the EM simulator. No parallel computation is applied in these experiments. All the time consumptions in the experiments are clock time.

A. Example 1

The first example is a WR-3 band (220–325 GHz) waveguide filter, which is composed of four coupled resonators operating in the TE101 mode. The filter has a Chebyshev response [30] (Fig. 4). Because of the fabrication methods in this frequency range, the filters can be complex in construction and difficult to design. The ranges of design variables are in Table I. The design specifications are that the passband is from 296 to 304 GHz (8-GHz passband centered at 300 GHz) and the max($|S_{11}|$) within the passband should be at least less than -20 dB and is as smaller as possible. The stopbands are from 280 to 292 GHz and from 308 to 320 GHz, where the max($|S_{11}|$) should be better than -1 dB. Therefore, the optimization problem is formulated as

$$\begin{aligned} &\text{minimize } \max(|S_{11}|), \quad 296\text{ GHz–}304\text{ GHz} \\ &\text{s.t. } \min(|S_{11}|) \geq -1\text{dB}, \quad 280\text{ GHz–}292\text{ GHz} \\ &\min(|S_{11}|) \geq -1\text{dB}, \quad 308\text{ GHz–}320\text{ GHz}. \end{aligned} \tag{16}$$

The initial design is obtained by coupling matrix fitting and is shown in Table II with a performance in Fig. 5(a). It can be seen that this response is far from the specifications. The Nelder–Mead simplex method [5] and the sequential quadratic programming method [4] are first used. These two methods are well-known local optimization methods, and many space mapping techniques are based on these two search engines. The implementation is based on MATLAB optimization toolbox functions fminsearch and fminimax. Each EM simulation costs about 2 min. The results are shown in Table III. It can be seen that both of them fail to find the narrow valley where the optimal design is located. It is not a surprise because the poor response of the initial design indicates that it is not near the optimal region, which is a major challenge for local optimization methods when facing filter design landscapes.

Ten runs with independent random numbers (including ten different initial populations) are carried out for SMEAFO to demonstrate the performance and the robustness. Because the initial response is far from the specifications and considering the ranges of the design variables, we set σ_{init} to 0.1. When the generated values by (14) are not within the ranges of the design variables, they are set to the nearest bound. As was said in Section III-D, the value of σ_{init} is a rough estimation

TABLE II

INITIAL SOLUTION AND AN OPTIMIZED SOLUTION (ALL SIZES IN MILLIMETERS) FOR EXAMPLE 1

Variables	$D12$	$D23$	Dt	$H12$	L	$L2$
Initial design	0.42	0.19	0.15	0.369	0.725	0.66
SMEAFO optimized design	0.442	0.257	0.168	0.296	0.735	0.658

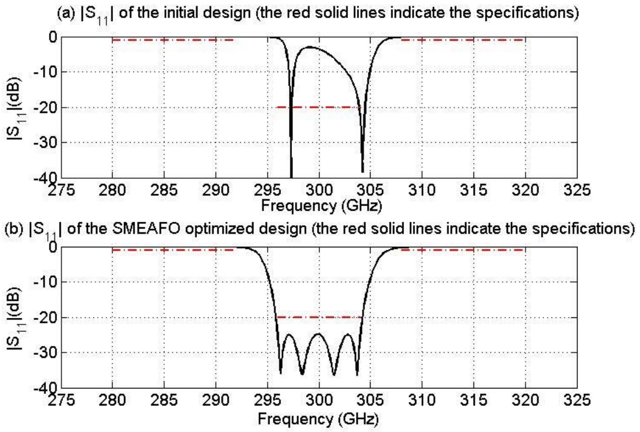


Fig. 5. Response of the waveguide filter.

TABLE III

OPTIMIZED RESULTS USING DIFFERENT METHODS FOR EXAMPLE 1

Methods	$max(\vert S_{11}\vert)$: 296–304GHz	$min(\vert S_{11}\vert)$: 280–292GHz	$min(\vert S_{11}\vert)$: 308–320GHz	time
Nelder-Mead simplex	–3.26 dB	–0.20 dB	–0.05 dB	9.2 hours
sequential quadratic programming	–4.22e-5 dB	–1.7e-3 dB	–0.84 dB	1.9 hours
DE	–22.86 dB	–0.08 dB	–0.17 dB	691.2 hours
SMEAFO	–24.85 dB	–0.12 dB	–0.25 dB	20.6 hours

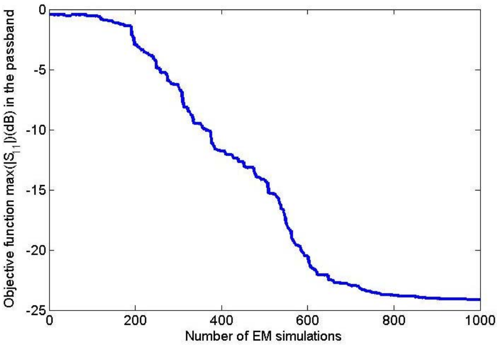


Fig. 6. SMEAFO convergence trend for Example 1 (average of ten runs).

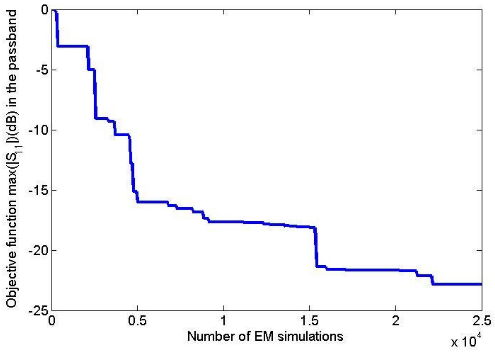


Fig. 7. DE convergence trend for Example 1.

and is not sensitive. The computing budget is 1000 EM simulations, but in most runs, convergence happens before 800 EM simulations. In all the ten runs using SMEAFO, the constraints are satisfied and the average objective function value is -24.14 dB. The best value is -26.90 dB, the worst value is -17.72 dB, and the standard deviation is 3.37. Eight out of ten runs obtain max($|S_{11}|$) (296–304 GHz) smaller than -24 dB. A medium one is provided in Tables II (optimized design variables) and III and Fig. 5(b) (performance). The convergence trend is shown in Fig. 6. It can be seen that the design quality is satisfactory using less than 600 EM simulations on average.

DE is also carried out using the initial population of the SMEAFO run provided in Tables II and III with the same related algorithm parameters of SMEAFO. The computing budget is set to 25 000 EM simulations. The obtained result is -22.86 dB. Hence, SMEAFO obtains more than 33 times speed improvement compared with standard DE for this example, making the unbearable time to be very practical (one month to one day) and obtaining an even better result. The convergence trend of DE is shown in Fig. 7. Comparing the two convergence trends, some observations can be made.

On average, the exploitation phase of SMEAFO starts from about 540 EM simulations with the starting performance of about -17 dB (according to the rule defined in Section III-C). The observations are as follows.

1) DE completes the exploration using about 8500 EM simulations (obtaining -17 dB), while SMEAFO uses about 540 EM simulations, obtaining about 16 times speed improvement.
2) DE then costs 16 500 EM simulations to improve the result from near-optimal designs to the final optimized design because of the rugged landscape in the valley, while the exploitation phase of SMEAFO costs about 250 EM simulations to get a better solution, achieving about 66 times speed improvement. This verifies the effectiveness of main ideas of SMEAFO in both phases.

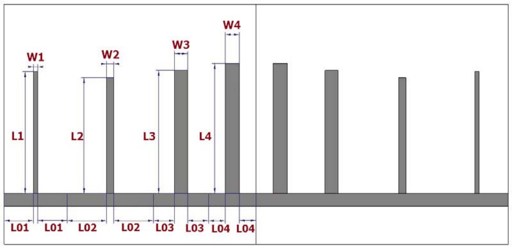


Fig. 8. Microstrip filter: front view.

TABLE IV
RANGES OF THE 12 DESIGN VARIABLES (ALL SIZES IN MILLIMETERS) FOR EXAMPLE 2

$Variables$	$W1$	$W2$	$W3$	$W4$	$L01$	$L02$
Lower bound	0.5	0.5	0.5	0.5	2	2
Upper bound	3.5	3.5	3.5	3.5	8	8
$Variables$	$L03$	$L04$	$L1$	$L2$	$L3$	$L4$
Lower bound	2	2	7	7	7	7
Upper bound	8	8	18	18	18	18

B. Example 2

The second example is an eighth-order microstrip filter working from 3.3 to 7.3 GHz, which is shown in Fig. 8. The ranges of design variables are in Table IV. As mentioned above, they are intentionally expanded to test SMEAFO in an extreme condition. The design specifications are that the passband is from 4 to 7 GHz and the stopbands are from 3.3 to 3.92 GHz and from 7.08 to 7.3 GHz. Therefore, the optimization problem is defined as follows:

$$\begin{aligned}&\text{minimize } \max(|S_{11}|), \quad 4\text{ GHz–7 GHz}\\&\text{s.t. } \min(|S_{11}|) \geq -3\text{dB}, \quad 3.3\text{ GHz–3.92 GHz}\\&\min(|S_{11}|) \geq -3\text{dB}, \quad 7.08\text{ GHz–7.3 GHz.}\end{aligned} \tag{17}$$

An equivalent circuit model is available, which is used for a first optimization to get the initial design, in which the simulation is performed by ADS circuit simulator (not Momentum). Because each ADS circuit simulation only costs a few seconds, standard DE is used. The optimized design variables (initial design for full-wave optimization) are shown in Table V with a performance in Fig. 9(a). It can be seen that the response of the optimized design using the equivalent circuit model is excellent in terms of circuit simulation, and when simulating it with the full-wave EM model, the response seems to be good as a starting point and only a slight move from the initial design is needed. However, this "correct slight move" is difficult.

TABLE V

INITIAL SOLUTION AND AN OPTIMIZED SOLUTION (ALL SIZES IN MILLIMETERS) FOR EXAMPLE 2

$Variables$	$W1$	$W2$	$W3$	$W4$	$L01$	$L02$
Initial design	0.591	0.746	1.476	1.324	4.440	5.408
$Variables$	$L03$	$L04$	$L1$	$L2$	$L3$	$L4$
Initial design	2.742	2.595	16.531	15.42	16.748	17.708
$Variables$	$W1$	$W2$	$W3$	$W4$	$L01$	$L02$
SMEAFO optimized design	0.549	0.953	1.727	1.887	3.885	5.245
$Variables$	$L03$	$L04$	$L1$	$L2$	$L3$	$L4$
SMEAFO optimized design	2.771	2.219	16.111	15.321	16.273	17.206

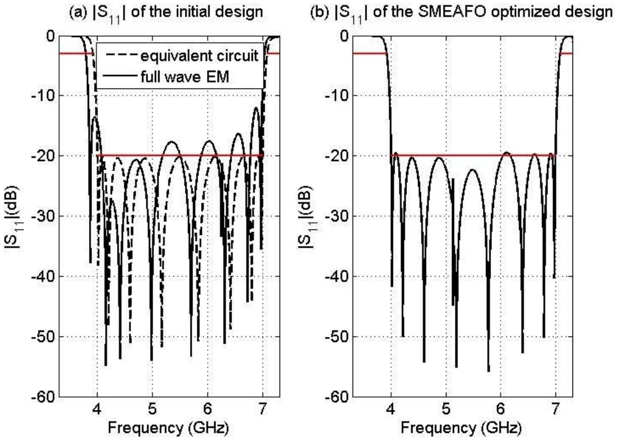


Fig. 9. Response of the microstrip filter.

TABLE VI

OPTIMIZED RESULTS USING DIFFERENT METHODS FOR EXAMPLE 2

Methods	$max(\lvert S_{11}\rvert)$: 4–7GHz	$min(\lvert S_{11}\rvert)$: 3.3–3.92GHz	$min(\lvert S_{11}\rvert)$: 7.08–7.3GHz	time
Nelder-Mead simplex	–6.29 dB	–2.81 dB	–3.00 dB	47.2 hours
sequential quadratic programming	–2.42 dB	–22.40 dB	–0.20 dB	77 hours
SMEAFO	–19.48 dB	–2.93 dB	–2.96 dB	96.9 hours

The Nelder–Mead simplex method and the sequential quadratic programming method are first applied. Note that these local optimization methods are not affected by the expanded search ranges, as a good starting point is available. For this example, each EM simulation costs about 3–6 min. The results are shown in Table VI. It can be seen that Nelder–Mead simplex fails to jump out of local optima, although the initial design is near the narrow valley, while sequential quadratic programming goes out of the narrow valley and is trapped in a local optimum in the outer region. Simulation data indicates that in all the eight test problems, this problem has the most rugged landscape.

Then, SMEAFO is carried out. Because of the intentionally set large ranges of the design variables, σ_{init} is set to 0.25. As was said in Section III-D, the value of σ_{init} is a rough estimation and is not sensitive. The computing budget is 1250 EM simulations. The results are shown in Tables V (optimized design variables) and VI and Fig. 9(b) (performance). The convergence trend is shown in Fig. 10. It can be seen that SMEAFO obtains a satisfactory result for this very challenging problem in terms of the ruggedness of the landscape and the narrowness of the optimal valley. This example shows that for problems that space mapping seems to be the suitable method, there are exceptions and SMEAFO can be a supplement for these exceptions.

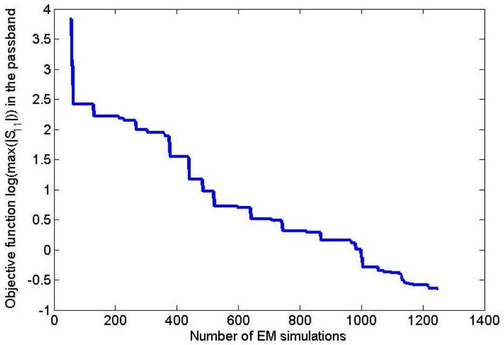


Fig. 10. SMEAFO convergence trend for Example 2.

Experiments on our real-world filter design test cases show that most hard filter optimization can be finished within 700 EM simulations using SMEAFO obtaining satisfactory results. For most test cases, the optimization time is one to two days. Note that SMEAFO is designed for filter optimization problems that may be difficult to solve by existing local optimization methods (space mapping-based methods without problem specific tuning often at most obtain the same solution quality compared with direct local search), comparing speed with such methods is therefore not relevant. Rather, an excellent result in a reasonable timeframe for hard problems is the goal of SMEAFO.

V. CONCLUSION

In this paper, the SMEAFO algorithm has been proposed. SMEAFO is aimed to serve as a widely applicable method (i.e., not restricted by filter types/responses) targeted at microwave filter optimization problems that are difficult to be solved by popular local optimization methods, while at the same time are not affordable to be solved by standard global optimization methods, so as to complement the state of the arts. Experiments show that SMEAFO can provide optimal filter designs that are comparable with the DE algorithm, which is expected to provide very high-quality design, but uses a reasonable timeframe and is several orders faster than DE. These results are achieved by our novel SAEA designed for filter landscapes, including the two-phase structure, the novel surrogate model-assisted search methods, and the training data selection method in each phase. In addition, SMEAFO with a lower fidelity EM model can be used to support space mapping, providing a good initial design and a database with lower fidelity model evaluation results. The SMEAFO-based filter design optimization tool can be downloaded from http://fde.cadescenter.com. Future works include developing parallelized SMEAFO.

ACKNOWLEDGMENT

The authors would like to thank Prof. S. Koziel, Reykjavik University, Iceland, for valuable discussions.

References

[1] M. Fakhfakh, E. Tlelo-Cuautle, and P. Siarry, *Computational Intelligence in Analog and Mixed-Signal (AMS) and Radio-Frequency (RF) Circuit Design*. Berlin, Germany: Springer, 2015.

[2] A. Deb, J. S. Roy, and B. Gupta, "Performance comparison of differential evolution, particle swarm optimization and genetic algorithm in the design of circularly polarized microstrip antennas," *IEEE Trans. Antennas Propag.*, vol. 62, no. 8, pp. 3920–3928, Aug. 2014.

[3] N. He, D. Xu, and L. Huang, "The application of particle swarm optimization to passive and hybrid active power filter design," *IEEE Trans. Ind. Electron.*, vol. 56, no. 8, pp. 2841–2851, Aug. 2009.

[4] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," *Acta Numer.*, vol. 4, pp. 1–51, Jan. 1995.

[5] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder–Mead simplex method in low dimensions," *SIAM J. Optim.*, vol. 9, no. 1, pp. 112–147, 1998.

[6] M. Sans, J. Selga, A. Rodríguez, J. Bonache, V. E. Boria, and F. Martín, "Design of planar wideband bandpass filters from specifications using a two-step aggressive space mapping (ASM) optimization algorithm," *IEEE Trans. Microw. Theory Techn.*, vol. 62, no. 12, pp. 3341–3350, Dec. 2014.

[7] S. Koziel, S. Ogurtsov, J. W. Bandler, and Q. S. Cheng, "Reliable space-mapping optimization integrated with EM-based adjoint sensitivities," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 10, pp. 3493–3502, Oct. 2013.

[8] X. Shang, Y. Wang, G. L. Nicholson, and M. J. Lancaster, "Design of multiple-passband filters using coupling matrix optimisation," *IET Microw., Antennas Propag.*, vol. 6, no. 1, pp. 24–30, 2012.

[9] J. W. Bandler, R. M. Biernacki, S. H. Chen, P. A. Grobelny, and R. H. Hemmers, "Space mapping technique for electromagnetic optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 42, no. 12, pp. 2536–2544, Dec. 1994.

[10] C. Zhang, F. Feng, V.-M.-R. Gongal-Reddy, Q. J. Zhang, and J. W. Bandler, "Cognition-driven formulation of space mapping for equal-ripple optimization of microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 7, pp. 2154–2165, Jul. 2015.

[11] Q. S. Cheng, J. W. Bandler, and S. Koziel, "Space mapping design framework exploiting tuning elements," *IEEE Trans. Microw. Theory Techn.*, vol. 58, no. 1, pp. 136–144, Jan. 2010.

[12] R. Ben Ayed, J. Gong, S. Brisset, F. Gillon, and P. Brochet, "Three-level output space mapping strategy for electromagnetic design optimization," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 671–674, Feb. 2012.

[13] M. A. E. Sabbagh, M. H. Bakr, and J. W. Bandler, "Adjoint higher order sensitivities for fast full-wave optimization of microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 8, pp. 3339–3351, Aug. 2006.

[14] T. J. Santner, B. J. Williams, and W. I. Notz, *The Design and Analysis of Computer Experiments*. Berlin, Germany: Springer, 2013.

[15] M. T. M. 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.

[16] J. E. Dennis and V. Torczon, "Managing approximation models in optimization," in *Multidisciplinary Design Optimization: State of the Art*. Hampton, VA, USA: SIAM, 1997, pp. 330–347.

[17] I. Couckuyt, A. Forrester, D. Gorissen, F. De Turck, and T. Dhaene, "Blind Kriging: Implementation and performance analysis," *Adv. Eng. Softw.*, vol. 49, pp. 1–13, Jul. 2012.

[18] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. New York, NY, USA: Springer-Verlag, 2005.

[19] A. Hoorfar, "Evolutionary programming in electromagnetic optimization: A review," *IEEE Trans. Antennas Propag.*, vol. 55, no. 3, pp. 523–537, Mar. 2007.

[20] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, Jul. 1999.

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

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

[23] Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, and K. Y. Lum, "Combining global and local surrogate models to accelerate evolutionary optimization," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 1, pp. 66–76, Jan. 2007.

[24] 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.

[25] B. Liu, D. Zhao, P. Reynaert, and G. G. E. Gielen, "Synthesis of integrated passive components for high-frequency RF ICs based on evolutionary computation and machine learning techniques," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 30, no. 10, pp. 1458–1468, Oct. 2011.

[26] C. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, 2007.

[27] J.-S. G. Hong and M. J. Lancaster, *Microstrip Filters for RF/Microwave Applications*, vol. 167. Hoboken, NJ, USA: Wiley, 2004.

[28] P. Rocca, G. Oliveri, and A. Massa, "Differential evolution as applied to electromagnetics," *IEEE Antennas Propag. Mag.*, vol. 53, no. 1, pp. 38–49, Feb. 2011.

[29] D. M. Himmelblau, *Applied Nonlinear Programming*. New York, NY, USA: McGraw-Hill, 1972.

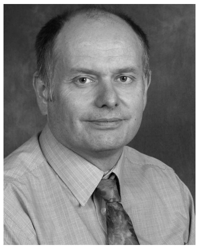
[30] X. Shang, M. Ke, Y. Wang, and M. J. Lancaster, "WR-3 band waveguides and filters fabricated using SU8 photoresist micromachining technology," *IEEE Trans. THz Sci. Technol.*, vol. 2, no. 6, pp. 629–637, Nov. 2012.

Bo Liu (M'15) received the B.S. degree from Tsinghua University, Beijing, China, in 2008, and the Ph.D. degree from MICAS Laboratories, University of Leuven, Leuven, Belgium, in 2012.

From 2012 to 2013, he was a Humboldt Research Fellow with the Technical University of Dortmund, Dortmund, Germany. In 2013, he joined Wrexham Glyndwr University, Wrexham, U.K., as a Lecturer, where he became a Reader of computer-aided design in 2016. He is currently an Honorary Fellow with the University of Birmingham, Edgbaston, Birmingham, U.K. He has authored or co-authored 1 book and more than 40 papers in international journals, edited books, and conference proceedings. His current research interests include design automation methodologies of analog/RF integrated circuits, microwave devices, MEMS, evolutionary computation, and machine learning.

Hao Yang was born in Wuhan, China, in 1991. He received the B.Eng. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, in 2014, and the B.Eng. degree in electronic and electrical engineering from the University of Birmingham, Edgbaston, Birmingham, U.K., in 2014, where he is currently pursuing the Ph.D. degree.

His current research interests include terahertz frequency filters and mutiplexers.



Michael J. Lancaster (SM'04) was born in West Yorkshire, U.K., in 1958. He received the bachelor's degree in physics and Ph.D. degree (with a focus on nonlinear underwater acoustics) from the University of Bath, Bath, U.K., in 1980 and 1984, respectively.

He was a Research Fellow with the Surface Acoustic Wave (SAW) Group, Department of Engineering Science, Oxford University, Oxford, U.K., where he was involved in the design of new, novel SAW devices, including RF filters and filter banks. In 1987, he became a Lecturer in electromagnetic theory and microwave engineering with the Department of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham, U.K., where he was involved in the study of the science and applications of high-temperature superconductors, mainly at microwave frequencies, and became the Head of the Department of Electronic, Electrical, and Systems Engineering in 2003. He has authored 2 books and over 190 papers in refereed journals. His current research interests include microwave filters and antennas, high-frequency properties and applications of a number of novel and diverse materials, and micromachining as applied to terahertz communications devices and systems.

Prof. Lancaster is a Fellow of the IET and the Institute of Physics, U.K. He is a Chartered Engineer and a Chartered Physicist. He has served on the IEEE MTT-S IMS Technical Committees.