Journal Article

An Efficient Method for Antenna Design Optimization Based on Evolutionary Computation and Machine Learning Techniques

Liu, B., Aliakbarian, H., Ma, Z., Vandenbosch, G., Gielen, G. and Excell, P.

An Efficient Method for Antenna Design Optimization Based on Evolutionary Computation and Machine Learning Techniques

Bo Liu, Hadi Aliakbarian, *Member, IEEE*, Zhongkun Ma, Guy A. E. Vandenbosch, *Fellow, IEEE*, and Georges Gielen, *Fellow, IEEE*

***Abstract*—In recent years, various methods from the evolutionary computation (EC) field have been applied to electromagnetic (EM) design problems and have shown promising results. However, due to the high computational cost of the EM simulations, the efficiency of directly using evolutionary algorithms is often very low (e.g., several weeks' optimization time), which limits the application of these methods for many industrial applications. To address this problem, a new method, called Surrogate Model Assisted Differential Evolution for Antenna Synthesis (SADEA), is presented in this paper. The key ideas are: (1) A Gaussian Process (GP) surrogate model is constructed on-line to predict the performances of the candidate designs, saving a lot of computationally expensive EM simulations. (2) A novel surrogate model-aware evolutionary search mechanism is proposed, directing effective global search even when a traditional high-quality surrogate model is not available. Three complex antennas and two mathematical benchmark problems are selected as examples. Compared with the widely used differential evolution and particle swarm optimization, SADEA can obtain comparable results, but achieves a 3 to 7 times speed enhancement for the antenna design optimization.**

***Index Terms*—Antenna design optimization, Antenna synthesis, Surrogate model assisted evolutionary algorithm, Gaussian process, Differential evolution, Efficient global optimization, Expensive black-box optimization**

I. Introduction

In recent years, antenna synthesis / design optimization through evolutionary algorithms (EAs) has been applied widely [1], [2], [3]. Because antenna synthesis often encounters multimodal black-box optimization problems [2], it falls into the playfield of evolutionary computation (EC) methods. At present, differential evolution (DE) [4] and particle swarm optimization (PSO) [5] are the most popular algorithms in the antenna synthesis area [1], [3]. Besides using canonical DE and PSO, various state-of-the-art methods in the EC field have been introduced and successfully applied, such as a self-adaptive DE [6] and the covariance matrix adaptation evolution strategy (CMA-ES) method [7]. In addition, multi-objective optimization of antennas [8], [9], [10] has been introduced.

Besides the current research focus of applying and improving EAs to generate better antenna designs, the research direction of efficiency enhancement is very important but has not been well investigated yet. Electromagnetic (EM) simulation is often a must in antenna synthesis, and the time needed to perform an EM simulation for a candidate solution varies from a few seconds to tens of minutes [11], or even several hours. The EA-based synthesizer often needs 30000-200000 evaluations (EM simulations) for antenna problems [11]. It is clear that drastically enhancing the efficiency without or with slightly sacrificing the performance is very significant for antenna synthesis. To the best of our knowledge, most of the current antenna synthesis methods accept the long computational time because EA is often the only possible method to obtain a very high-quality design. Nevertheless, a few pioneer research works have been conducted concentrating on efficiency enhancement. They can be classified into the following categories:

- Space mapping [12], [13]
  In efficient design optimization (synthesis) methods based on space mapping, a relatively computationally cheap coarse model (e.g., an EM simulation model with coarse meshes) is constructed and optimized, in order to approximately locate the optimal subregion of the design space. A surrogate model will then be constructed there, which will be used to optimize the microwave component. The intermediate optimal solutions will be verified by expensive fine (high-fidelity) EM simulations and the obtained simulation data will update the surrogate model. The main limitation of such methods is that the success of the space mapping mechanism comes from the basic assumption that the optimal point of the coarse and the fine models are not far away from each other in the design space. However, this is not always true in practical antenna design and often cannot be known beforehand. Detailed analysis and experiments are in [14].
- Cooperation with EM simulation [15], [16]
  For some structures, the speed of optimization can highly be increased by only solving a single electromagnetic computation problem for the whole optimization process. In such cases, with the help of Maxwell's equations, a matrix equation is obtained which relates the optimization variables and the fitness function. The optimization technique is applied on this equation to find the global optimum. The application of these techniques is somewhat limited due to restrictions on the variables and fitness

Manuscript received November 23rd, 2012. Bo Liu is with Department of Computer Science, Technische Universitaet Dortmund, Dortmund, Germany. Georges Gielen, Hadi Aliakbarian, Zhongkun Ma and Guy A. E. Vandenbosch are with ESAT, Katholieke Universiteit Leuven, Leuven, Belgium (e-mail: bo.liu@cs.tu-dortmund.de, {Bo.Liu, Hadi. Aliakbarian, Zhongkun.Ma, Guy.Vandenbosch, Georges.Gielen}@esat.kuleuven.be, liu_bo765@yahoo.com.cn).

function definition.

- Improving the search engine [7]
  A typical method is illustrated in [7], which introduces the CMA-ES method into the antenna design optimization field. The main idea is to apply EAs which are able to obtain a suitable solution with a reduced number of exact function evaluations (computationally expensive EM simulations). Although more computational overhead may be necessary for the evolution operations, the efficiency can still considerably be enhanced since EM simulations dominate the optimization time. This method is general and up to 1.6 times speed enhancement has been shown compared to the canonical PSO. Nevertheless, a more considerable speed enhancement is needed in many cases.
- Surrogate model assisted evolutionary algorithm [10], [11], [17]
  The surrogate model assisted evolutionary algorithm (SAEA) has been introduced into the antenna design optimization (synthesis) area in [10], [11], [17]. Unlike the surrogate model in the space mapping-based methods, the surrogate modeling and the evolution operations work in a cooperative manner in SAEAs [14]. [17] and [11] use the efficient global optimization (EGO) method [18] and the ParEGO [19] method from the computational intelligence field for the single- and multi-objective optimization of antennas, respectively. Gaussian Process (GP) machine learning [20] and the expected improvement (EI) [18] prescreening are used. A new method is proposed in [10] which integrates the GP machine learning and the generation control-based method [21] to avoid incorrect convergence. Considerable speed enhancement has been shown in these methods. Although they are significant contributions to the efficiency enhancement of antenna synthesis, the dimensionality that can be handled by these methods is not large (i.e., at most 8) [10], [11], [17]. This is caused by the "curse of dimensionality" in the computational intelligence field. More details will be provided in Section II and Section IV. On the other hand, for practical antenna synthesis, the design parameters can vary from a few to more than 30, which poses a big challenge for the present SAEA-based synthesis methods.

In this paper, a new efficient method is proposed for antenna synthesis, called Surrogate Model Assisted Differential Evolution for Antenna Synthesis (SADEA). This is a Gaussian Process-based SAEA for computationally expensive black-box global optimization problems. To address the "curse of dimensionality", a new evolutionary search mechanism and the corresponding surrogate modeling method are proposed, whose goal is to conduct effective and efficient global optimization for 10- to 30-dimensional problems when a very limited number of EM simulations are allowed. DE is selected as the search engine in SADEA, but the proposed SAEA mechanism is also compatible with PSO, the genetic algorithm (GA) and the evolution strategies (ES). The SADEA method aims to:

- provide highly optimized results comparable to directly using a standard EA with EM simulations in the optimization loop;
- use much less computational effort compared with using a standard EA with EM simulations, and as such make the computation time of the synthesis much more practical;
- be general enough for most antenna synthesis problems, especially for complex antenna synthesis problems with around 20 to 30 design parameters.

The remainder of the paper is organized as follows. Section II briefly reviews the background and the challenges encountered in SAEA. Section III introduces the basic techniques used in SADEA, the DE algorithm and the GP machine learning. Section IV introduces the SADEA method. Section V tests the SADEA method on three practical antenna examples with 7, 10 and 19 parameters, respectively, as well as 30-dimensional mathematical benchmark problems. The comparisons are performed to DE, PSO and two SAEAs. Concluding remarks are presented in Section VI.

II. Background of Surrogate Model Assisted Evolutionary Algorithms

SAEA is a recent approach for dealing with computationally expensive optimization problems. These algorithms employ surrogate models to replace the computationally expensive function evaluations (e.g., high-fidelity EM simulations). Since the construction of the surrogate model and its use to predict the function values cost much less effort than directly embedding the computationally expensive function evaluator (e.g., EM simulator) within the optimizer, the computational cost can significantly be reduced. In recent years, many surrogate modeling methods and the corresponding SAEAs were investigated. Among them, Gaussian Process or Kriging, response surface methods, artificial neural networks (ANN), support vector machines and radial basis function models show good performances and are widely used.

As has been said above, the surrogate modeling and the evolutionary algorithm must work in a cooperative manner in SAEA, instead of training the surrogate model in a pre-processing phase (i.e., an off-line surrogate model). Earlier methods, such as some examples in [22], do not consider the model uncertainty in the evolution process. After a number of samples with exact function evaluations, a surrogate model is constructed and the predicted optimal points are evaluated by exact function evaluations. The surrogate model is then updated and the above steps are repeated until the convergence is achieved. To address the issue of incorrect convergence of the above method, individual and generation control-based methods are introduced to consider the estimation error [21]. The goal is to determine when the surrogate model must be updated (using more exact function evaluations) to avoid false optima. These are the prediction-based methods.

Meanwhile, prescreening is introduced into SAEA (e.g., using the EI, or the lower confidence bound (LCB)) [18], [23]. Instead of expecting the surrogate model prediction to replace the exact function evaluation in the prediction-based methods (the prediction uncertainty should be as small as possible to that end), prescreening methods aim to select the possible promising candidates from the newly generated candidate solutions utilizing the prediction uncertainty. Because

both the EA and the prescreening methods contribute to the global search, prescreening-based methods can often detect the globally or near globally optimal solutions efficiently for small-scale problems. Note that most prescreening methods, such as the EI, are specially defined for the GP model, because they rely on the Gaussian distribution of the outputs.

However, the "curse of dimensionality" is a common problem to both kinds of SAEA. Some pioneer research works on medium-scale (20- to 30-dimensional) SAEA are [24], [25], [26]. For typical mathematical benchmark problems, a reasonably good result can often be obtained by the methods of [26] and [24], but they cost 6000 to 8000 exact function evaluations, which are often too many for practical use. In [25], promising results have been obtained for some benchmark problems using 1000 exact function evaluations, but for others the results are not good enough, especially for multimodal problems (i.e., problems having more than one local optimum). To summarize, much effort is required to develop both effective and efficient SAEAs for medium-scale problems, which are adequate for the requirements of practical antenna synthesis.

III. Basic Techniques

The EA and the surrogate modeling method are the two key components of an SAEA. In SADEA, the DE algorithm is selected as the search engine and the GP machine learning is selected as the surrogate modeling method. In the following, brief introductions to these techniques will be provided.

A. GP machine learning

GP machine learning [20] assumes that the objective function is a sample of a Gaussian stochastic process. The distribution of the function value of a new point can be predicted by the available points. Both a predicted value and a prediction uncertainty are provided for a new point, and can be used by the prescreening methods to select promising newly generated candidates to perform (expensive) exact function evaluations. We will now explain this.

Suppose that there are n training data $x = (x_1, x_2, \ldots, x_n)$ and their corresponding exact function values are $y = (y_1, y_2, \ldots, y_n)$. Using the GP model with the correlation function $Corr(x_i, x_j)$, the function value $y(x^*)$ at a new point x^* can be predicted as

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

where

$$\begin{aligned} &R_{i,j} = Corr(x_i, x_j), i, j = 1, 2, \ldots, n \\ &r = [Corr(x^*, x_1), Corr(x^*, x_2), \ldots, Corr(x^*, x_n)] \\ &\hat{\mu} = (I^T R^{-1} I)^{-1} I^T R^{-1} y \end{aligned} \tag{2}$$

There are different correlation functions to describe the correlation between two candidate designs. Eqn. (3) shows an example of a correlation function:

$$\begin{aligned} &Corr(x_i, x_j) = exp(-\textstyle\sum_{l=1}^{d} \theta_l |x_i^l - x_j^l|^{p_l}) \\ &\theta_l > 0, 1 \leq p_l \leq 2 \end{aligned} \tag{3}$$

where d is the dimension of x and θ_l is the correlation parameter which determines how fast the correlation decreases when x_i^l moves in the l direction. p_l is related to the smoothness of the function with x^l. To determine the parameters θ_l and p_l, GP machine learning maximizes the likelihood function:

$$h = \frac{1}{(2\pi)^{n/2}(\sigma^2)^{n/2}|R|^{1/2}} exp(-\frac{(y - I\mu)^T R^{-1}(y - I\mu)}{2\sigma^2}) \tag{4}$$

where I is a $n \times 1$ vector of ones. Assuming that θ_l and p_l are known (the variables in the likelihood function are just μ and σ), by setting the derivatives of the likelihood function to 0, the value of μ must be $\hat{\mu}$ from eqn. (2) and the value of σ^2 must be

$$\hat{\sigma}^2 = (y - I\hat{\mu})^T R^{-1}(y - I\hat{\mu}) n^{-1} \tag{5}$$

By substituting $\hat{\mu}$ and $\hat{\sigma}^2$ into eqn. (4), the likelihood function only depends on θ_l and p_l. Then, eqn. (4) can be maximized to obtain the estimates of $\hat{\theta}_l$ and $\hat{p}_l$. R and r are known based on the obtained correlation function.

The prediction uncertainty $\hat{s}^2$ based on the best linear unbiased prediction, which is used to assess the model accuracy, is described as:

$$\hat{s}^2(x^*) = \hat{\sigma}^2[I - r^T R^{-1} r + (I - r^T R^{-1} r)^2 (I^T R^{-1} I)^{-1}] \tag{6}$$

Several prescreening methods can be used to measure the "potential" of a candidate design based on the predicted value (eqn. (1)) and the prediction uncertainty (eqn. (6)). Here we introduce the LCB prescreening, which is then used in SADEA. Considering a minimization problem,

$$y_{lcb}(x) = \hat{y} - \omega \hat{s}(x), \omega \in [0, 3] \tag{7}$$

where ω is a constant. With this prescreening, $y_{lcb}(x)$, instead of $\hat{y}(x)$ itself, is used to measure the quality of x. The LCB prescreening balances the search between promising areas (i.e., with low $\hat{y}(x)$ values) and less explored areas (i.e., with high $\hat{s}(x)$ values). Following the suggestion in [23], $\omega = 2$ is used for balancing the exploration and exploitation of LCB in our method.

B. The DE algorithm

The DE algorithm outperforms many other EAs in terms of solution quality and convergence speed [4] for optimization in a continuous space. DE uses a simple differential operator to create new candidate solutions and a one-to-one competition scheme to greedily select new candidates. The DE algorithm and its improvements have widely been used in antenna synthesis. In the following, a brief introduction to DE will be provided.

The i^{th} candidate solution in the d-dimensional search space at generation t can be represented as

$$x_i(t) = [x_{i,1}, x_{i,2}, \ldots, x_{i,d}] \tag{8}$$

At each generation t, the mutation and crossover operators are applied to the candidate solutions, and a new population arises. Then, selection takes place, and the corresponding candidate solutions from both populations compete to comprise the next generation. The operators are:

Mutation:

$$V_i(t+1) = x_{best}(t) + F(x_{r_1}(t) - x_{r_2}(t)) \\ r_1, r_2 \in \{1, \ldots, NP\}, r_1 \neq r_2 \neq i \quad (9)$$

where $V_i(t+1)$ is a mutant counterpart in the $t+1$ generation. Indices r_1 and r_2 are randomly chosen and mutually different, and also different from the current index i. NP is the population size. $x_{best}(t)$ is the best individual of the current population. F is the scaling factor. In this work, we use a constant F.

There are several kinds of DE mutations, and the mutation in eqn. (9) is called DE/best/1 strategy. The advantage is that beneficial information can be shared more effectively, because the current best individual participates in the generation of all the $V_i(t+1)$. However, its drawback is that the diversity of the generated population is lower. Another widely used DE mutation is shown in eqn. (10), called the DE/rand/1 strategy. Compared to eqn. (9), the only difference is that the current best candidate, $x_{best}(t)$, is replaced by a randomly selected candidate $x_{r_3}(t)$ from the current population. Using this strategy, the convergence speed is slower than using the DE/best/1 strategy, but the diversity of the population is enhanced, avoiding premature convergence.

$$V_i(t+1) = x_{r_3}(t) + F(x_{r_1}(t) - x_{r_2}(t)) \\ r_1, r_2, r_3 \in \{1, \ldots, NP\}, r_1 \neq r_2 \neq r_3 \neq i \quad (10)$$

Crossover: a trial vector U is generated as follows:

$$U_i(t+1) = [U_{i,1}(t+1), U_{i,2}(t+1), \ldots, U_{i,d}(t+1)] \quad (11)$$

$$U_{i,j}(t+1) = \begin{cases} V_{i,j}(t+1), \text{if}(rand(i,j) \leq CR) | j = randn(i) \\ x_{i,j}(t), \text{otherwise} \end{cases} \quad (12)$$

where $rand(i,j)$ is an independent random number uniformly distributed in the range $[0,1]$. Parameter $randn(i)$ is a randomly chosen index from the set $\{1, \ldots, d\}$. Parameter $CR \in [0,1]$ is a constant parameter called the crossover rate.

Selection:

$$x_i(t+1) = \begin{cases} U_i(t+1), \text{if } f(U_i(t+1)) < f(x_i(t)) \\ x_i(t), \text{otherwise} \end{cases} \quad (13)$$

where the function f is the objective function, i.e., the function to be minimized. The candidate solution, $x_i(t+1)$, becomes a candidate solution in the new population. Then, the next iteration begins. For more details about the DE algorithm, please see [4].

IV. The SADEA Algorithm

A. Analysis of the problem and motivations

By developing SAEAs, the high optimization ability of EA and the computational efficiency of the surrogate model can be integrated to achieve the first two requirements from Section I for small-scale problems. But medium and large-scale optimization problems (e.g., complex antenna synthesis problems with 10-30 design variables, the third requirement) remain a challenge even for the state-of-the-art methods in the computational intelligence field, hence the need for SADEA.

The success of an SAEA is highly affected by having a high quality surrogate model. For prediction-based SAEAs, the prediction error of the newly generated candidates should be small enough to approximately replace the exact function evaluations. For prescreening-based SAEAs, a reasonably good ranking needs to be obtained by the surrogate model and the prescreening method. In either case, the quality of the surrogate model is determined by several factors.

One of them is the surrogate modeling mechanism. GP machine learning is used in SADEA. The reason is that GP machine learning is a theoretically principled method for determining a much smaller number of control parameters compared to other surrogate modeling approaches [27], [28]. For some other kernel-based methods, such as ANN, overfitting occurs in many cases because the learning mechanism is sensitive to the control parameters (e.g., number of neurons in the hidden layer, the type of the transfer function). However, the method to determine the control parameters is not mathematically sound till now and much experience has to be used. In addition, as the uncertainty for each predicted point is provided in the GP model, various presceening methods are available, which provides additional help for SAEA. Therefore, GP surrogate modeling is selected in SADEA.

Another important factor determining the quality of the surrogate model is the number of training data. It is intuitive that more sampling data (training data) are needed for medium- and high-dimensional problems to construct a reasonably good surrogate model: the higher the dimensionality, the more training data are necessary. However, it has to be noticed that the training data can only be provided by exact function evaluations, which is computationally expensive. This illustrates why available state-of-the-art SAEAs in the computational intelligence field either need a lot of function evaluations to achieve reasonably good results, or use less exact function evaluations but with suboptimal results. The former case is at odds with the second requirement of SADEA and the latter case is at odds with the first requirement of SADEA.

B. Key ideas of SADEA

To address this problem, the key idea of SADEA is to investigate and to improve the cooperation between the surrogate modeling and the EA. Most available SAEAs are based on standard EAs (e.g., the standard GA [24], [26], or the standard $\mu + \lambda$ ES [25]), which need complex replacements in the selection operation. Since the newly generated candidates often spread in different areas of the search space, in order to make the replacements in these different subregions being reliable, an accurate surrogate model for various kinds of data (various subregions) is necessary. This can only be achieved by using more exact function evaluations, which serve as the training data or which are directly used for comparison and replacement. SADEA, on the other hand, proposes a new surrogate model-aware search mechanism collaborating with a simplified GP model.

Acknowledging that the traditional high-quality surrogate model covering various subregions to assist standard EAs is difficult to construct with limited training data, a special kind

of surrogate model is needed and should be appropriately used by the new search mechanism. In other words, the surrogate modeling-aware search mechanism has three requirements: (1) the search mechanism should be able to achieve global optimization; (2) the surrogate model should be able to support the correct selection or replacement in the search; (3) the number of exact function evaluations should be as small as possible. In SADEA, we construct a reasonably accurate GP model only for the top-ranked candidate solutions among the newly generated candidates in each iteration. In the search, we also use the top-ranked candidates (in practice, the best candidates based on prescreening) for exact function evaluation, which is just supported by the surrogate model. Only one exact function evaluation is used in each iteration.

Although only using a limited number of training data, the above special surrogate model can be achieved if the training data are near the top-ranked candidates in the group of newly generated candidates. Considering that the GP model construction and prediction are based on the correlation function (eqn. (3)), the available training data around the candidate to be predicted contribute more than the training data far away from it. For this reason, we select the latest generated optimal candidates (except those iterations which still use initial samples) as the training data. Assuming that optimal solutions have a similar structure and that the fitness values of the candidate solutions are improving, the latest optimal candidates can be considered as being near to the newly generated good candidates and can provide a good surrogate model for them. Note that the quality of the GP model constructed in this way may be poor when predicting or ranking lower-ranked candidates. Therefore, we only use the best candidate by prescreening in order to increase the probability that the prescreening-based "best" candidate is also top-ranked in reality. The detailed algorithm will be described in the next subsection.

C. The general framework of SADEA

Assume that our optimization problem is to minimize $y = f(x)$ and $x \in [a, b]^d$ (a negative sign can be added for maximization). Like most other SAEAs, SADEA records all the exactly evaluated solutions and their function values in a database. The database is constructed by the values x and their corresponding function values y. Once an exact function evaluation is carried out for a new x, the x and its exact (not predicted) function value y will be stored in the database. For the initialization of the database, a Design of Experiments (DOE) method, Latin Hypercube sampling (LHS) [29] is used. The LHS sampling method samples the design space more uniformly, and hence, needs fewer samples to achieve a more effective sampling. LHS and other DOE methods have widely been used for the database initialization in SAEAs [18]. The SADEA algorithm works as follows:

Step 1: Use the Latin Hypercube sampling to generate α solutions from $[a, b]^d$, evaluate all the individuals using exact function evaluations and then use them to form the initial database.

Step 2: If a preset stopping criterion is met (e.g., a maximum number of exact function evaluations), output the best solution from the database; otherwise go to step 3.

Step 3: Select the λ best solutions (i.e., with the lowest function values) from the database to form a population P. Update the best solution obtained so far.

Step 4: Apply the DE mutation (eqn. (9) or (10)) and crossover (eqn. (12)) operators on P to generate λ child solutions.

Step 5: Take the τ newest solutions from the database (i.e., the last τ solutions from the database) and their function values as the training data to build a GP surrogate model.

Step 6: Prescreen the λ child solutions generated in Step 4 by using the GP model from Step 5 with the LCB prescreening.

Step 7 : Evaluate (exact function evaluation) the prescreened best child solution from Step 6, add this evaluated solution and its function value to the database. Go back to Step 2.

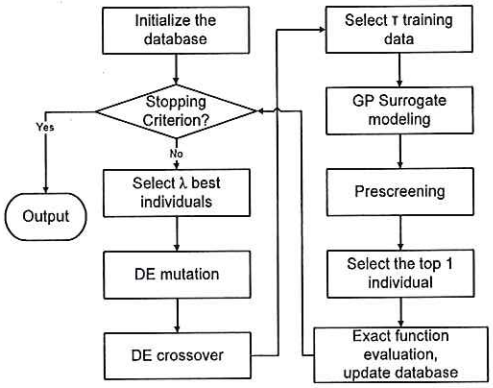

Fig. 1. The flow diagram of SADEA

The flow diagram of SADEA is shown in Fig.1.

In SADEA, the DE mutation and crossover operators are used to generate new candidates, but a very different search framework is proposed in order to cooperate with the surrogate modeling. Two types of mutation are introduced in Section III, the DE/best/1 mutation and the DE/rand/1 mutation. According to Section III (B), the advantage of the DE/best/1 mutation is to generate optimal solutions more efficiently, while for the DE/rand/1 mutation, premature convergence can often effectively be prevented because of the high diversity of the generated solutions. Empirical studies show that in most cases the diversity of the population is enough when using the DE/best/1 mutation. However, in some cases, the design parameters must be an integer or discrete number, which may be enforced by the meshing and the EM simulation tools. For such cases, the DE/rand/1 mutation is necessary. Examples are given in Section V. Besides the mutation operator, quantization methods [4] need to be used. In all the DE search operators, floating numbers are always used, while these floating numbers are only rounded to the nearest allowed values in the function evaluation when necessary. More details are in [4]. The goal

of the DE/rand/1 mutation and the quantization method is to maintain the diversity and the exploration ability as much as possible when the diversity is lost to some extent due to the rounding. However, the advantage of the DE algorithm is the global optimization in a continuous space, instead of integer programming [30]. For such problems, we recommend the P-SO or the ant colony algorithm (ACO). Note that this surrogate model-aware search framework, which is the main contribution of SADEA, is general and compatible with different kinds of EAs (e.g., PSO, ES, ACO) as well as with many state-of-the-art EA-based methods for antenna synthesis.

D. Parameter settings of SADEA

1) Parameter setting rules: There are several control parameters in SADEA. In the following, some guiding rules based on references and empirical investigations are provided.

- DE parameters (the scaling factor F and the crossover rate CR): The setting of the DE parameters has been well investigated in [4]. We suggest to set F to 0.8, and CR to 0.8, which are used in all the experiments in Section V. Empirical studies show that highly optimized results can often be obtained with a good convergence speed using this setting. For some problems with quite flat overall shapes (e.g., Rastrigin function), the value of CR can be decreased to obtain highly optimized results.
- Parameters in surrogate modeling: The ooDACE toolbox [31] is used. The number of training data τ is decided by the trade-off between the surrogate model quality and the surrogate model training time. We suggest to use 80 to 120 for τ, which results in a reasonably good surrogate model and an efficient GP model training. In the experiments, $\tau = 100$ is used.
- LCB parameter: ω needs to be set when using LCB. We use $\omega = 2$ as suggested by [25], [23].
- The parameters in the search framework: The number of initial samples α is suggested to be set between $3 \times d$ to $5 \times d$. The population size λ is suggested to be set between 40 and 60 for 10- to 30-dimensional problems. In the following, $\lambda = 50$ is used for problems with 10 or more variables, and $\lambda = 40$ is used for problems with less than 10 variables.

2) Discussions: In the following, some experimental results are provided to clarify some parameter setting rules. Mathematical benchmark problems are used (see Appendix) and the statistics are based on 20 runs.

We first analyze the setting of F and α. The 10-dimensional Ackley function is used as the test bench, which is a highly multimodal problem (with numerous local optima). 700 function evaluations are used. According to the above parameter setting rules, we use $F = 0.8$ and $\alpha = 40$ as the reference setting (the other parameters follow the above rules), which provides the reference result of SADEA. We then revise F and α for comparisons. In each experiment, only a single parameter is revised. The results are shown in Table I. It can be seen that: (1) Using a small scaling factor F is not a good choice for SADEA, because the exploration ability will decrease. (2)

TABLE I
STATISTICS OF THE SADEA RESULTS FOR DIFFERENT F AND α SETTINGS

Settings	best	worst	average	std
Ref	7.05e-5	1.57e-4	1.19e-4	1.28e-4
$F = 0.5$	6.3e-5	2.58	0.65	0.93
$F = 0.6$	9.4e-5	7.35	0.85	2.31
$\alpha = 30$	0	1.81e-4	9.16e-5	7.79e-5
$\alpha = 50$	8.97e-5	1.59e-4	1.20e-4	2.07e-5

TABLE II
STATISTICS OF THE SADEA RESULTS FOR DIFFERENT λ SETTINGS

Settings	best	worst	average	std
Ref	1.24e-4	0.036	0.014	0.01
$\lambda = 30$	4.26e-6	0.62	0.13	0.21
$\lambda = 70$	4.1e-3	0.51	0.13	0.17

Good results can be obtained using $3 \times d$ to $5 \times d$ initial samples, which validates the empirical rules.

We then analyze the setting of λ. The 15-dimensional Griewank function is used as the test bench, which is a highly multimodal problem. 800 function evaluations are used. Because there are 15 decision variables, $\lambda = 50$ is used as the reference setting. For other parameters, besides those with a recommended setting of a single value, α is set to 70. λ is then revised to 30 and 70. The results are shown in Table II. It can be observed that the results of using $\lambda = 30$ and $\lambda = 70$ are worse than that of the reference setting $\lambda = 50$, although they are still good for this problem. When observing the convergence curve, it was found that when using $\lambda = 30$, 40% of the runs got premature convergence, while using $\lambda = 70$, 50% of the runs show that the convergence is not achieved yet after 800 evaluations, so $\lambda = 50$ is a good compromise.

V. EXPERIMENTAL RESULTS AND COMPARISONS

In this section, the SADEA algorithm is demonstrated for the optimization of three complex antennas with increasing simulation cost and two mathematical benchmark problems. A 60GHz on-chip wireless antenna (10 design parameters), a four-element linear array antenna (19 parameters) and a two-dimensional antenna array (7 parameters) are selected. ADS-Momentum is used as the EM simulator for the first example and Magmas [32], [33] is used as the EM simulator for the second and the third example. Because of the meshing accuracy required by Magmas, all the variables in the second example must be integers and the grids are set to $0.2mm$ and $0.5mm$ for the third example [34]. The bounds of the design variables are set both by the design rules of the technology used and the experience of the designer. One evaluation of a candidate solution costs about 4 minutes for the first example, 2 to 4.5 minutes for the second example and 11-15 minutes for the third example, respectively. For the first example, SADEA stops when the performance cannot be improved for 30 consecutive iterations. For the second example, the synthesis time is restricted to a night's time (about 12 hours) and we restrict the number of EM simulations to 600 in the experiment. For the third example, we restrict the synthesis to

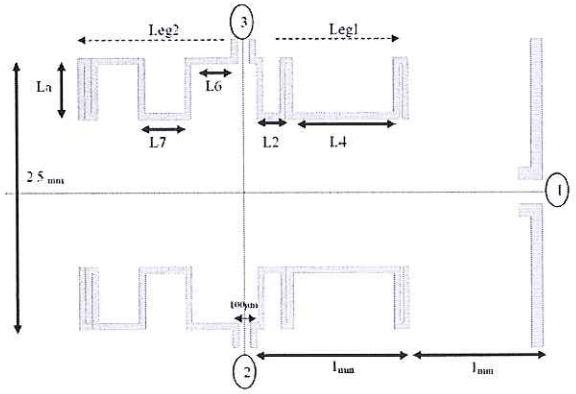

Fig. 2. Proximity coupling scheme and final layout of the optimized antennas (antenna example 1)

be finished in three days and at most 350 EM simulations are allowed. The first example is run on a PC with Intel 2.4GHz Xeon CPU with 12GB RAM, the second example is run on a PC with 2.0GHz CPU with 128GB RAM and the third example is run on a work station with an Intel(R) I7 2.8GHz CPU with 4GB RAM. All the time consumptions reported in the experiments are clock time. To test the effectiveness of SADEA for tackling the "curse of dimensionality', two 30-dimensional multimodal benchmark problems [24] are used in Section V(D). Various widely used methods are compared with SADEA in the following subsections.

A. Example 1: Inter-chip wireless antenna

Inter-chip antenna design is a recent popular antenna problems [35]. With the rapid growth in high-frequency integrated circuit technology, the new method of wireless inter-chip communication is proposed as an alternative solution with some advantages compared to wired chip interconnection [36].

This example analyzes a problem of short-range communication between three antennas at 60GHz, shown in Fig. 2. In this inter-chip communication scheme on a 90nm CMOS silicon technology as the substrate, the antenna 2 communicates with the antenna 3, and both of them receive interferences from the antenna 1 which is a fixed wideband dipole. The antennas 2 and 3 are decided to be meander-line dipoles. The goal of the optimization is to maximize the coupling from antenna 2 to antenna 3 and at the same time to reduce the crosstalk from antenna 1. Therefore the optimization problem is defined as follows:

$$\begin{aligned} &\text{maximize} \quad coupling(antenna2, antenna3) \\ &s.t. \quad crosstalk(antenna1, antenna2) \leq -30dB \end{aligned} \tag{14}$$

The distance between antenna 2 and antenna 3 is $2.5mm$ (see Fig. 2). In order to make sure that the crosstalk from antenna 1 to both antenna 2 and 3 is the same, antenna 2 and 3 are mirrored. However, both of them can be asymmetrical as shown in Fig. 2. Each meander line antenna has 5 horizontal sections on each of the arms, namely $L1$ to $L5$ on one arm

TABLE III
VALUES OF THE VARIABLES OF THE OPTIMIZED ANTENNAS 2 AND 3 (μm) FOR ANTENNA SYNTHESIS EXAMPLE 1

$L1$	$L2$	$L3$	$L4$	$L5$
8.34	136.80	8.34	823.33	23.19
$L6$	$L7$	$L8$	$L9$	$L10$
328.85	328.85	328.85	6.76	6.69

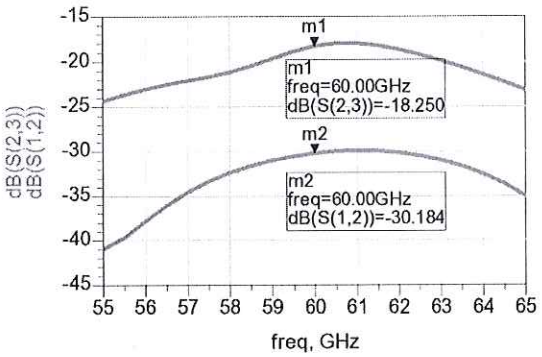

Fig. 3. Simulated coupling from antenna 2 to antenna 3 and the simulated crosstalk from antenna 1, optimized as a function of frequency at 60GHz by SADEA

and $L6$ to $L10$ on the other arm. Thus, another constraint is the total length of each arm which is fixed to be $1mm$, as stated in eqn. (15), which can be handled in the algorithm's data generation. The minimum values of these variables are bounded by the selected technology.

$$\sum_{m=1}^{5} L_m = \sum_{m=6}^{10} L_m = 1mm \tag{15}$$

The value of the antenna width La is $500\mu m$. The dipole length of antenna 1 is $2.6mm$ and the dipole is situated at $1mm$ from the other antennas.

In SADEA, the initial number of samples is set to 40 (because there are 10 design variables) and all the other parameters are shown in Section IV(D). The layouts of the optimized antennas are shown in Fig. 2, and the obtained values of the 10 design parameters are shown in Table III. The coupling between antenna 2 and 3 is optimized to -18.25 dB while the constraint is satisfied with -30.18 dB crosstalk. Fig. 3 shows the simulation details.

The total number of exact evaluations of SADEA is 302, costing 21 hours. The comparison method we have selected for the first example is the selection-based differential evolution (SBDE) algorithm [37], which is widely used in single-objective constrained optimization (including its variants). The SBDE method for the same example takes 2000 iterations to converge, with a computational time of 5.9 days. The optimized performances are coupling (antenna 2, antenna 3)= -18.84dB and crosstalk (antenna 1, antenna 2)=-30.30dB. It can be seen that the performances of the optimized antennas by SADEA and SBDE are comparable, but SBDE consumes 7 times more computational effort than SADEA.

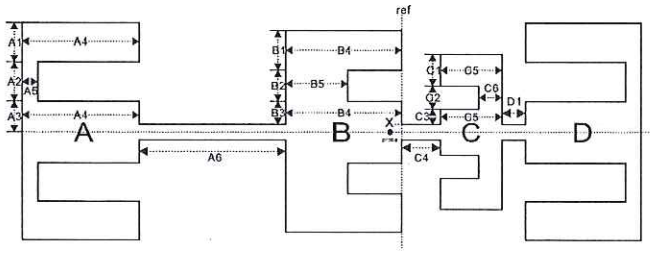

Fig. 4. Four-element antenna array top view (antenna example 2)

TABLE IV
RANGES OF THE 19 DESIGN VARIABLES (ALL SIZES IN MM) FOR ANTENNA SYNTHESIS EXAMPLE 2

Variables	*A1*	*A2*	*A3*	*A4*	*A5*	*A6*	*B1*	*B2*	*B3*	*B4*
Lower bound	4	4	4	12	4	4	6	4	4	20
Upper bound	12	12	12	30	12	28	20	16	8	40
Variables	*B5*	*C1*	*C2*	*C3*	*C4*	*C5*	*C6*	*D1*	*X*	
Lower bound	4	4	4	4	2	12	4	2	5	
Upper bound	16	12	12	12	26	30	12	24	12	

B. Example 2: Four-element linear array antenna

A highly compact low-cost and strongly coupled four-element linear array antenna [38], [39] is chosen as the second example in this subsection. This antenna has been optimized by the Particle Swarm Optimization (PSO) algorithm [40], [38]. We therefore use PSO with the same settings as in [40] but using the fitness function defined in this subsection as the comparison reference. The latter obtains a better result compared to [40]. The goal is to maximize the realized gain in the operating frequency range from 3.4GHz to 3.8GHz. In this band, the S11 parameter should be below -10dB and the gain should at least be 13dB. The substrate used is FR4. The performances at five equidistant frequency points (3.4GHz, 3.5GHz, 3.6GHz, 3.7GHz and 3.8GHz) are evaluated. Therefore the optimization problem is defined as follows:

$$\begin{aligned} &\text{maximize} \quad \textstyle\sum_{i=1}^{5} RG_D \\ &s.t. \quad RG_D \geq 12.54dB \end{aligned} \tag{16}$$

where $RG_D = gain - 10 \times log10(1 - Reflection^2)$ and $Reflection = 10^{S11/20}$.

The topology of the four-element antenna is shown in Fig. 4 and the shape of the antenna is controlled by 19 design parameters indicated in the figure. The ranges of the design parameters are shown in Table IV.

For this example, 10 runs are carried out for both SADEA and PSO. Thus, the consumed computational time for SADEA is about 10 hours to 14 hours, which conforms to the time constraint. In all the 10 runs using SADEA, the constraints are satisfied and the average objective function value is 71.05dB using 600 EM simulations. The variables of the best result of the 10 runs are shown in Table V and the simulated performance is shown in Fig. 5. Compared to PSO (using the settings from [40]), about 1700 EM simulations are needed to obtain a comparable average objective function value as SADEA, so the speed enhancement of SADEA compared to PSO is nearly 3 times for this example. Using PSO, the

TABLE V
THE SYNTHESIZED 19 DESIGN VARIABLES (BEST RESULT) OBTAINED BY SADEA FOR ANTENNA SYNTHESIS EXAMPLE 2

Variables	*A1*	*A2*	*A3*	*A4*	*A5*	*A6*	*B1*	*B2*	*B3*	*B4*
Values	12	12	12	30	4	28	20	4	4	28
Variables	*B5*	*C1*	*C2*	*C3*	*C4*	*C5*	*C6*	*D1*	*X*	
Values	4	12	4	4	26	30	12	24	8	

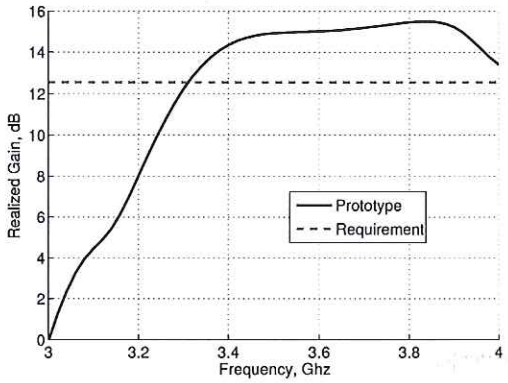

Fig. 5. Realized gain of the antenna synthesized by SADEA (best result) for antenna example 2

average objective function value after 3000 EM simulations is 71.65dB. The convergence curve of SADEA is shown in Fig. 6. It can be seen that the convergence tendency is very clear, so it can be expected that the already good result will further be improved when allowing more EM simulations. Note that this example is an integer programming problem, but the search engine of SADEA, the DE algorithm, is good at continuous optimization problems. Considering that the surrogate model-aware search framework in SADEA is also compatible with search engines (e.g., PSO, ACO) good at mixed integer-discrete optimization problems, better performances can be expected from such combinations, which is a future research direction.

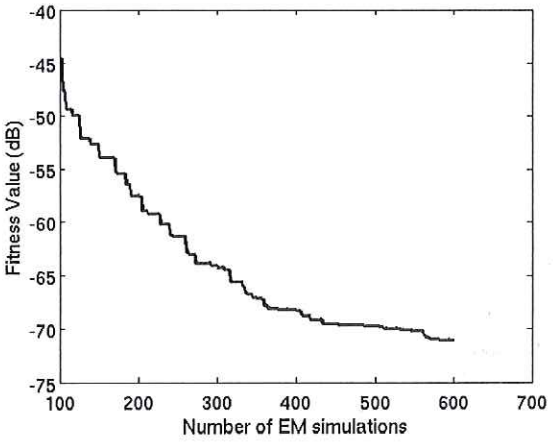

Fig. 6. SADEA convergence trend for the antenna synthesis example 2 in 600 EM simulations (average of 10 runs)

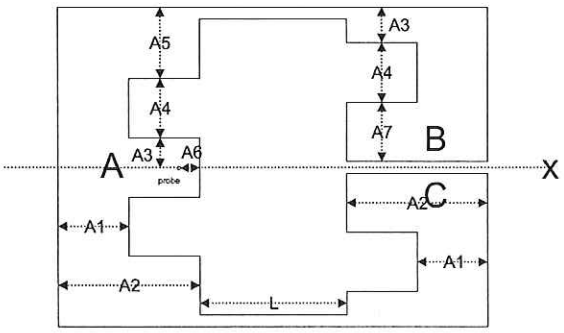

Fig. 7. Topology of the two-dimensional antenna array

Besides PSO, we compare SADEA with a widely used generation control-based SAEA [21], which is a popular CPU-time reduction technique for expensive optimization problems. With any EA, that method uses exact evaluations (in antenna synthesis, these are EM simulations) for some generations, and surrogate model predictions in other generations, which is controlled adaptively according to the prediction error. In [21], an ANN model is used, while the same GP model as in SADEA is used here. DE with the same parameters as in SADEA is also used for fair comparison. In 10 runs, all the constraints are satisfied, and the average objective function value is 70.82dB, using about 1450 EM simulations. It can therefore be observed that SADEA has clear advantages in terms of efficiency compared to the reference method for this example.

C. Example 3: Two-dimensional array

A two-dimensional array [34] is selected as the third example. The topology and the design parameters (seven of them are independent) are depicted in Fig. 7. To date, this topology has the highest aperture efficiency for a medium-sized medium bandwidth planar array (more than 10% bandwidth) as reported in literature [34]. This topology was optimized using PSO in [34]. The same objective function is used here, which is as follows:

$$\begin{aligned}
&\text{minimize} \quad \textstyle\sum_{i=1}^{5} M_i + \sum_{i=1}^{5} R_i \\
&S_i = max(VSWR_i, VSWR_{iD}) \\
&L_i = max(13dB - Gain_i, 0) \\
&C = \forall VSWR_i \leq VSWR_{iD} \text{ AND } \forall Gain_i \geq 13dB \\
&R_i = \begin{cases} VSWR_i, \text{if } C = \text{True} \\ S_i, \text{otherwise} \end{cases} \\
&M_i = \begin{cases} 13dB - Gain_i, \text{if } C = \text{True} \\ L_i, \text{otherwise} \end{cases} \\
&i = 1, 2, \ldots, 5.
\end{aligned} \tag{17}$$

where $VSRW_i$ and $Gain_i$ are the VSRW and the broadside gain values at the selected five frequency points, i.e., 2.33GHz, 2.40GHz, 2.45GHz, 2.50GHz and 2.57GHz. The setting of the specification $VSWR_{iD}$ is quite complex and the details are in [34]. For comparison, we use exactly the same parameter ranges and grids, the same EM solver, the same simulation method and fitness function as in [34].

TABLE VI
STATISTICS OF THE SADEA RESULTS FOR THE 30-d ACKLEY AND GRIEWANK TEST PROBLEMS

Problem	best	worst	average	std
Ackley	1.9491	4.9640	3.0105	0.9250
Griewank	0.7368	1.0761	0.9969	0.1080

Five runs are carried out for SADEA, PSO and DE, respectively. The parameters of SADEA and DE are as described in Section IV(D) and the PSO parameters are the same as those in [34]. The average objective function value of SADEA is 9.26 using 320 EM simulations. 1126 EM simulations are needed when using PSO to achieve a comparable result. It can be observed that about a 3.5 times speed enhancement can be obtained, thus decreasing the 1.5 weeks computational time taken by PSO down to less than 3 days. After 2500 EM simulations (about 20 days) using PSO, the average objective function value is 9.19. Therefore, the result obtained by SADEA within 3 days is of high quality. When applying DE, 1388 EM simulations are needed to obtain a comparable result with SADEA (average over 5 runs). Compared to DE, more than a 4 times speed enhancement is achieved by SADEA for this example.

D. Benchmark problem tests

In SAEA research, mathematical benchmark problems are often used to evaluate an algorithm. In this subsection, the Ackley function and the Griewank function (see Appendix) are used, and the dimensionality is selected to be 30. Note that they are highly multimodal complex problems. The global optimum of these two test problems is 0. 20 runs are performed for each problem and 1000 exact function evaluations are used. In SADEA, the initial number of samples is set to 100 (because there are 30 design variables) and all the other parameters have been shown in Section IV(D). The performance of SADEA is shown in Table VI. It can be seen that the benchmark problems are largely optimized by SADEA, and the optimized solutions are already near the global optimum in only 1000 exact evaluations.

It is interesting to compare SADEA with the state-of-the-art SAEAs for medium-scale problems. In the experiments of a state-of-the-art method, GS-SOMA [24], these two benchmark problems are also used. Table VII shows the comparison result between SADEA and GS-SOMA. The two comparison items are: (1) The average function value that GS-SOMA achieves in 1000 exact function evaluations (the third column of Table VII). This value is estimated from the figures showing the convergence trends in [24]. (2) The number of exact function evaluations GS-SOMA requires to achieve the same result as SADEA achieves in 1000 function evaluations (the fourth column of Table VII). In [24] GS-SOMA uses 8000 exact function evaluations and the final results after 8000 exact evaluations are shown by tables. As the final result of the first test problem is worse than the SADEA result, we write >8000. For the second test problem, this number is estimated

TABLE VII
COMPARISON OF SADEA WITH GS-SOMA [24] ON COMMON TEST FUNCTIONS

Common Problem	SADEA (1000)	GS-SOMA (1000)	GS-SOMA (number)	speedup
Ackley	3.0105	20	>8000	>8
Griewank	0.9969	365	2500	2.5

from the figures showing the convergence trends in [24]. The average values obtained by each method are compared.

From Table VII, it can be seen that for 30-dimensional (medium-scale) complex problems, SADEA can achieve a considerable speed enhancement even compared to the state-of-the-art SAEA in the computational intelligence field.

VI. CONCLUSIONS

In this paper, the surrogate model assisted differential evolution algorithm for efficient antenna synthesis (SADEA) has been presented. Experimental results show that SADEA can achieve approximately 3 to 7 times efficiency enhancement while providing highly optimized design solutions comparable to standard evolutionary algorithms for antenna synthesis. Therefore, the three goals of this work (high optimization ability, high efficiency and general for complex problems up to 20-30 dimensions) are achieved. SADEA addresses the problem of medium-scale computationally expensive global optimization, which is especially useful for on-chip and complex antenna design optimization. This is achieved by the key idea of the presented surrogate model-aware evolutionary search mechanism. Future works will focus on integrating a variety of search engines into the developed surrogate model-aware search mechanism and extending SADEA to multi-objective synthesis of antennas.

APPENDIX

Benchmark functions:

A. Ackley Problem

$$\begin{aligned} &min \quad f(x) = -20e^{-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}} - e^{\frac{1}{d}\sum_{i=1}^{d} cos(2\pi x_i)} \\ &x \in [-32.768, 32.768], i = 1, \ldots, d \\ &minimum : f(x^*) = 0 \end{aligned} \tag{18}$$

B. Griewank Problem

$$\begin{aligned} &min \quad f(x) = 1 + \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} cos(\frac{x_i}{\sqrt{i}}) \\ &x \in [-600, 600], i = 1, \ldots, d \\ &minimum : f(x^*) = 0 \end{aligned} \tag{19}$$

REFERENCES

[1] A. Hoorfar, "Evolutionary programming in electromagnetic optimization: a review," *Antennas and Propagation, IEEE Transactions on*, vol. 55, no. 3, pp. 523–537, 2007.

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

[3] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *Antennas and Propagation, IEEE Transactions on*, vol. 52, no. 2, pp. 397–407, 2004.

[4] K. Price, R. Storn, and J. Lampinen, *Differential evolution: a practical approach to global optimization*. Springer-Verlag New York Inc, 2005.

[5] M. Clerc, *Particle swarm optimization*. Wiley-ISTE, 2010.

[6] S. Goudos, K. Siakavara, T. Samaras, E. Vafiadis, and J. Sahalos, "Self-adaptive differential evolution applied to real-valued antenna and microwave design problems," *Antennas and Propagation, IEEE Transactions on*, vol. 59, no. 4, pp. 1286–1298, 2011.

[7] M. Gregory, Z. Bayraktar, and D. Werner, "Fast optimization of electromagnetic design problems using the covariance matrix adaptation evolutionary strategy," *Antennas and Propagation, IEEE Transactions on*, vol. 59, no. 4, pp. 1275–1285, 2011.

[8] N. Jin and Y. Rahmat-Samii, "Advances in particle swarm optimization for antenna designs: Real-number, binary, single-objective and multiobjective implementations," *Antennas and Propagation, IEEE Transactions on*, vol. 55, no. 3, pp. 556–567, 2007.

[9] S. Goudos, Z. Zaharis, D. Kampitaki, I. Rekanos, and C. Hilas, "Pareto optimal design of dual-band base station antenna arrays using multi-objective particle swarm optimization with fitness sharing," *Magnetics, IEEE Transactions on*, vol. 45, no. 3, pp. 1522–1525, 2009.

[10] B. Liu, H. Aliakbarian, S. Radiom, G. Vandenbosch, and G. Gielen, "Efficient multi-objective synthesis for microwave components based on computational intelligence techniques," in *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012, pp. 542–548.

[11] M. John and M. Ammann, "Antenna optimization with a computationally efficient multiobjective evolutionary algorithm," *Antennas and Propagation, IEEE Transactions on*, vol. 57, no. 1, pp. 260–263, 2009.

[12] S. Koziel and S. Ogurtsov, "Numerically efficient design optimization of a printed 2.45 ghz yagi antenna," in *Antennas and Propagation (EuCAP), 2010 Proceedings of the Fourth European Conference on*. IEEE, 2010, pp. 1–3.

[13] J. Bandler, Q. Cheng, S. Dakroury, A. Mohamed, M. Bakr, K. Madsen, and J. Sondergaard, "Space mapping: the state of the art," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 52, no. 1, pp. 337–361, 2004.

[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," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 30, no. 10, pp. 1458–1468, 2011.

[15] A. Babakhani, J. Lavaei, J. Doyle, and A. Hajimiri, "Finding globally optimum solutions in antenna optimization problems," in *Antennas and Propagation Society International Symposium (APSURSI), 2010 IEEE*. IEEE, 2010, pp. 1–4.

[16] J. Johnson and Y. Rahmat-Samii, "Genetic algorithms and method of moments (ga/mom) for the design of integrated antennas," *Antennas and Propagation, IEEE Transactions on*, vol. 47, no. 10, pp. 1606–1614, 1999.

[17] E. Siah, M. Sasena, J. Volakis, P. Papalambros, and R. Wiese, "Fast parameter optimization of large-scale electromagnetic objects using direct with kriging metamodeling," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 52, no. 1, pp. 276–285, 2004.

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

[19] J. Knowles, "Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 50–66, 2006.

[20] T. Santner, B. Williams, and W. Notz, *The design and analysis of computer experiments*. Springer, 2003.

[21] Y. Jin, M. Olhofer, and B. Sendhoff, "A framework for evolutionary optimization with approximate fitness functions," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 5, pp. 481–494, 2002.

[22] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, vol. 9, no. 1, pp. 3–12, 2005.

[23] J. Dennis and V. Torczon, "Managing approximation models in optimization," *Multidisciplinary design optimization: State-of-the-art*, pp. 330–347, 1997.

[24] D. Lim, Y. Jin, Y. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 3, pp. 329–355, 2010.

[25] M. Emmerich, K. Giannakoglou, and B. Naujoks, "Single-and multi-objective evolutionary optimization assisted by gaussian random field metamodels," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 4, pp. 421–439, 2006.

[26] Z. Zhou, Y. Ong, P. Nair, A. Keane, and K. Lum, "Combining global and local surrogate models to accelerate evolutionary optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 1, pp. 66–76, 2007.

[27] E. Ackermann, J. de Villiers, and P. Cilliers, "Nonlinear dynamic systems modeling using gaussian processes: Predicting ionospheric total electron content over south africa," *J. Geophys. Res*, vol. 116, p. A10303, 2011.

[28] D. Buche, N. Schraudolph, and P. Koumoutsakos, "Accelerating evolutionary algorithms with gaussian process fitness function models," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 35, no. 2, pp. 183–194, 2005.

[29] M. Stein, "Large sample properties of simulations using latin hypercube sampling," *Technometrics*, pp. 143–151, 1987.

[30] J. Gao, H. Li, and Y. Jiao, "Modified differential evolution for the integer programming problems," in *Artificial Intelligence and Computational Intelligence, 2009. AICI'09. International Conference on*, vol. 1. IEEE, 2009, pp. 213–219.

[31] D. Gorissen, I. Couckuyt, P. Demeester, T. Dhaene, and K. Crombecq, "A surrogate modeling and adaptive sampling toolbox for computer based design," *The Journal of Machine Learning Research*, vol. 11, pp. 2051–2055, 2010.

[32] G. Vandenbosch and A. Van de Capelle, "Mixed-potential integral expression formulation of the electric field in a stratified dielectric medium-application to the case of a probe current source," *Antennas and Propagation, IEEE Transactions on*, vol. 40, no. 7, pp. 806–817, 1992.

[33] Y. Schols and G. Vandenbosch, "Separation of horizontal and vertical dependencies in a surface/volume integral equation approach to model quasi 3-d structures in multilayered media," *Antennas and Propagation, IEEE Transactions on*, vol. 55, no. 4, pp. 1086–1094, 2007.

[34] Z. Ma and G. A. Vandenbosch, "Low-cost wideband microstrip arrays with high aperture efficiency," *Antennas and Propagation, IEEE Transactions on*, vol. 60, no. 6, pp. 3028–3034, 2012.

[35] J. Lin, H. Wu, Y. Su, L. Gao, A. Sugavanam, J. Brewer *et al.*, "Communication using antennas fabricated in silicon integrated circuits," *Solid-State Circuits, IEEE Journal of*, vol. 42, no. 8, pp. 1678–1687, 2007.

[36] K. Kim, B. Floyd, J. Mehta, H. Yoon, C. Hung, D. Bravo, T. Dickson, X. Guo, R. Li, N. Trichy *et al.*, "On-chip antennas in silicon ics and their application," *Electron Devices, IEEE Transactions on*, vol. 52, no. 7, pp. 1312–1323, 2005.

[37] K. Zielinski and R. Laur, "Constrained single-objective optimization using differential evolution," in *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2006, pp. 223–230.

[38] Z. Ma, V. Volski, and G. Vandenbosch, "Optimal design of a highly compact low-cost and strongly coupled 4 element array for wlan," *Antennas and Propagation, IEEE Transactions on*, vol. 59, no. 3, pp. 1061–1065, 2011.

[39] V. Volski, P. Delmotte, and G. Vandenbosch, "Compact low-cost 4 elements microstrip antenna array for wlan," in *Wireless Technology, 2004. 7th European Conference on*. IEEE, 2004, pp. 277–280.

[40] Z. Ma and G. Vandenbosch, "Comparison of weighted sum fitness functions for pso optimization of wideband medium-gain antennas," *Radioengineering*, vol. 21, no. 1, p. 505, 2012.

Bo Liu received the B.S. degree from Tsinghua University, P. R. China, in 2008. He received his Ph.D. degree at the MICAS laboratories of the Katholieke Universiteit Leuven, Belgium, in 2012, under the supervision of Prof. Dr. Georges Gielen. Since October 2012, he is a Humboldt research fellow and is working with Technical University of Dortmund and Technical University of Munich, Germany. His research interests lie in evolutionary computation, machine learning, fuzzy logic and design automation methodologies of analog / RF integrated circuits and antennas. He has authored or coauthored 1 book and more than 20 papers in international journals and conference proceedings. He was a technical consultant of Accelicon Technologies Inc. He was a session organizer on computationally expensive optimization and optimization in uncertain environments in IEEE World Congress on Computational Intelligence 2012. He is in the editorial board or as a reviewer in artificial intelligence and electronic design automation fields, such as IEEE Transactions on Evolutionary Computation, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Journal of Engineering. He is also a book reviewer of Springer, Elsevier and Bentham Science Publishers.

Hadi Aliakbarian (M'00) received the B.S and M.S degrees in Electrical and Telecommunication Engineering from the University of Tehran in 2002 and 2005, respectively. He worked for the microwave laboratory and the Center of Excellence on Applied Electromagnetics at the University of Tehran as an associated researcher from 2005 to 2007. He joined the Katholieke Universiteit Leuven, more specifically the TELEMIC research group of the Department of Electrical Engineering, in 2007 and is working towards a Ph.D. in Electrical Engineering. His research interests include different aspects of digital and analogue beam steering (especially in space conditions), small antennas, UWB antennas, antenna optimization, and EMC problems.

Zhongkun Ma received his 1st M.S. degree, 2nd M.S degree and doctor degree, in Information and Communication Techniques from Groep T, in Electrical Engineering from KU Leuven and in Electrical Engineering from KU Leuven, all in Belgium, in 2007, 2009 ands 2013, respectively. His research interests are in the area of radiation, microwave and nano-antennas, and optimization techniques.

Guy A. E. Vandenbosch received the M.S. and Ph.D. degrees in Electrical Engineering from the Katholieke Universiteit Leuven, Leuven, Belgium, in 1985 and 1991, respectively. From 1991 to 1993, he held a postdoctoral research position at the Katholieke Universiteit Leuven. Since 1993, he has been a Lecturer, and since 2005, a Full Professor at the same university. Guy Vandenbosch has teached or teaches courses on Electromagnetic Waves, Antennas, Electromagnetic Compatibility, Electrical Engineering, Electronics, and Electrical Energy, and Digital Steer- and Measuring Techniques in Physics. His research interests are in the area of electromagnetic theory, computational electromagnetics, planar antennas and circuits, nano-electromagnetics, EM radiation, EMC, and bio-electromagnetics. His work has been published in ca. 150 papers in international journals and has been presented in ca. 250 papers at international conferences.

Georges G. E. Gielen received his M.Sc. and Ph.D. degrees in Electrical Engineering from the Katholieke Universiteit Leuven, Belgium, in 1986 and 1990, respectively. In 1990, he was appointed as a postdoctoral research assistant and visiting lecturer at the department of Electrical Engineering and Computer Science of the University of California, Berkeley. In 1993, he was appointed assistant professor at the Katholieke Universiteit Leuven, where he was promoted to full professor in 2000. His research interests are in the design of analog and mixed-signal integrated circuits, and especially in analog and mixed-signal CAD tools and design automation (modeling, simulation and symbolic analysis, analog synthesis, analog layout generation, analog and mixed-signal testing). He has authored or coauthored two books and more than 400 papers in edited books, international journals and conference proceedings. He received the 1995 Best Paper Award in the John Wiley international journal on Circuit Theory and Applications, and was the 1997 Laureate of the Belgian Royal Academy on Sciences, Literature and Arts in the discipline of Engineering. He received the 2000 Alcatel Award from the Belgian National Fund of Scientific Research for his innovative research in telecommunications, and won the DATE 2004 Best Paper Award. He is a Fellow of the IEEE, served as elected member of the Board of Governors of the IEEE Circuits And Systems (CAS) society and as chairman of the IEEE Benelux CAS chapter. He served as the President of the IEEE Circuits And Systems (CAS) Society in 2005.