Journal Article

GASPAD: A General and Efficient mm-wave Integrated Circuit Synthesis Method Based on Surrogate Model Assisted Evolutionary Algorithm

Liu, B., Zhao, D. and Gielen, G.

GASPAD: A General and Efficient mm-wave Integrated Circuit Synthesis Method Based on Surrogate Model Assisted Evolutionary Algorithm

Bo Liu, Dixian Zhao, Patrick Reynaert and Georges Gielen, *Fellow, IEEE*

***Abstract*—The design and optimization (both sizing and layout) of mm-wave integrated circuits (ICs) have attracted much attention due to the growing demand in industry. However, available manual design and synthesis methods suffer from a high dependence on design experience, being inefficient or not general enough. To address this problem, a new method, called general mm-wave IC synthesis based on Gaussian process model assisted differential evolution (GASPAD), is proposed in this paper. A medium-scale computationally expensive constrained optimization problem must be solved for the targeted mm-wave IC design problem. Besides the basic techniques of using a global optimization algorithm to obtain highly optimized design solutions and using surrogate models to obtain a high efficiency, a surrogate model-aware search mechanism (SMAS) for tackling the several tens of design variables (medium scale) and a method to appropriately integrate constraint handling techniques into S-MAS for tackling the multiple (high-) performance specifications are proposed. Experiments on two 60GHz power amplifiers in a 65nm CMOS technology and two mathematical benchmark problems are carried out. Comparisons with the state-of-art provide evidence of the important advantages of GASPAD in terms of solution quality and efficiency.**

***Index Terms*—mm-wave integrated circuit design automation, high-frequency integrated circuit, RF circuit synthesis, surrogate model assisted evolutionary computation, expensive optimization, Gaussian process**

I. Introduction

In recent years, design and optimization methodologies for mm-wave frequency integrated circuits (IC) are attracting more and more attention. This trend will continue in the foreseeable future, since the demand for high-data-rate wireless communications is constantly increasing [1]. However, mm-wave IC design still highly depends on the designer's experience. The design procedure is often time consuming and often gets sub-optimal results. Two important reasons for this are:

- The equivalent circuit models of integrated passive components (e.g., inductor, transformer), which are critical in radio-frequency (RF) ICs, are narrow-band models. They are not sufficient for mm-wave circuit simulation where the distributed effects of the passive components have to be taken into account. As a result, the designers are forced to rely on experience, intuition and time consuming electromagnetic (EM) simulators to predict the circuit performance and revise the design parameters. The design procedure involves quite a number of iterations and is time consuming even for experienced designers.
- The traditional mm-wave IC design method relies on a systematic step by step design procedure, but it is sometimes difficult to optimize the desired circuit performance. Take for example the design of power amplifier (PA). Most mm-wave PA designs optimize the saturated output power (P_{sat}) and consequently, the maximum power added efficiency ($PAE@P_{sat}$). However, optimizing the PAE at the 1dB compression point (P_{1dB}) is more important to have a high average efficiency when transmitting modulated signals (e.g., 16QAM) [2]. Nevertheless, it is not easy to find out the optimal load impedance (by load-pull simulation) and the optimal bias point to optimize $PAE@P_{1dB}$ by the traditional PA design method.

In contrast, RF IC synthesis follows the "simulation and optimization" problem solving automation method [3]. The above challenges can therefore be addressed. To that end, powerful and efficient design automation techniques for mm-wave ICs are needed.

Obtaining highly optimized design solutions (effectiveness) in a practical time (efficiency) is the common requirement of IC synthesis methods. In literature, the high-frequency IC synthesis research can be divided into three subareas with different requirements on effectiveness and efficiency:

- low-GHz RF IC synthesis
  Some successful research works are available in this area [3], [4], [5], [6], [7] and the main focus is the effectiveness, or the optimality. The novelty is computationally cheap parasitic-aware models for passive components are generated and are used for simulation. Like many analog circuit sizing methods, evolutionary algorithms (EAs) are used to obtain highly optimized design solutions. Mathematically, they solve a constrained optimization problem, assuming that the number of simulations is not a limitation. A few hundreds of additional simulations can be used for obtaining solutions with better quality. In contrast, the following two subareas also try to provide high-

This research was partly supported by the Humboldt research fellowship, Germany.

B. Liu is with the Department of Computing, Glyndwr University, Wrexham, U.K. (e-mail: b.liu@glyndwr.ac.uk, liubo168@gmail.com).

D. Zhao, P. Reynaert and G. Gielen are with ESAT-MICAS, Katholieke Universiteit Leuven, Leuven, Belgium (e-mail: Dixian.Zhao@esat.kuleuven.be, Patrick.Reynaert@esat.kuleuven.be, Georges.Gielen@esat.kuleuven.be).

quality solutions to constrained optimization problems, but with the assumption that the number of simulations is limited, which is restricted by the synthesis time.

- mm-wave IC synthesis focusing on small-signal performance optimization
  The authors proposed the first synthesis method for mm-wave ICs working at 100GHz or above, called efficient machine learning-based differential evolution (EMLDE) [8]. Due to the f_T of many technologies (e.g., 90nm, 65nm CMOS), maximizing the power gain (small-signal performance) is often the main consideration for ICs working at 100GHz or above. In this area, besides using an EA to achieve highly optimized design solutions, efficiency becomes the main challenge, since at such frequencies computationally expensive EM simulation is unavoidable. When directly embedding the EM simulation into the EAs, an impractically long optimization time will result [8]. A surrogate model assisted evolutionary algorithm (SAEA) was introduced into mm-wave IC synthesis in EMLDE. A Gaussian Process (GP) surrogate model was employed to replace many expensive EM simulations. A challenge in SAEA research is the "curse of dimensionality". When an optimization problem has 20 to 30 design variables or more, the efficiency enhancement obtained by traditional SAEA may reduce considerably [9], [10]. Nevertheless, having 20 to 30 design variables is common in mm-wave IC synthesis. A decomposition method exploiting the properties of the targeted problem was proposed for dimension reduction in EMLDE. A disadvantage of EMLDE, however, is its somewhat complicated implementation.
- General mm-wave IC synthesis
  To the best of our knowledge, there is no available method for general mm-wave IC synthesis. Again, efficiency is the bottleneck. For one thing, the computationally expensive EM simulations are unavoidable. For another thing, it is difficult to apply the decomposition method for dimension reduction used in EMLDE. Indeed, EMLDE relies on the stage-by-stage design method for mm-wave amplifiers focusing on small-signal performance optimization. Maximizing the power gain (G_p) can be considered separately for each stage. However, both large-signal and small-signal performances need to be considered for general mm-wave IC synthesis. For example, for a 60GHz PA, the PAE, P_{1dB} and G_p all need to be maximized. A stage designed for gain maximization may not be a good design for efficiency maximization. When using the decomposition method from EMLDE, appropriate specifications of all performance metrics for each stage are a must, but this is even not easy to specify for well-experienced designers. In addition, because of the multiple (high-performance) specifications, good constraint handling techniques are highly needed, instead of the static penalty function method in EMLDE that is only suitable for not tight S-parameter constraints.

This paper proposes the first general method for mm-wave IC synthesis, which considers both large-signal and small-signal performances. Mathematically, the mm-wave IC synthesis problem targeted in this paper is a medium-scale (15-50 dimensional) computationally expensive constrained optimization problem. As said above, to gain generality, the circuit must be considered as a whole and the decomposable structure is lost, introducing again the "curse of dimensionality". An SAEA technique for medium-scale expensive optimization is still an open area in the computational intelligence field. Moreover, handling constraints in SAEA has not been investigated well either. To address these problems, a new method, called General mm-wave IC Synthesis Based on Gaussian Process Model Assisted Differential Evolution (GASPAD), is proposed. The GASPAD method aims to:

- develop a general mm-wave IC synthesis method starting from a given circuit topology, performance specifications and some hints on layout (e.g., the metal layer to be used, the transistor layout template with different numbers of fingers), without any initial design nor the individual specifications of each stage;
- provide highly optimized results (including both objective function optimization and the satisfaction of multiple tight constraints) comparable to the results obtained by directly using a widely used EA-based constrained optimization method with EM simulations embedded, which is often the best synthesis method with respect to the solution quality;
- use much less computational effort compared with using the above reference method, and as such make the computation time of the synthesis practical.

The remainder of this paper is organized as follows. Section II introduces the motivations and basic techniques. Section III presents the GASPAD method, including its main ideas, the general framework and the parameter settings. Section IV tests GASPAD on two 60GHz 65nm PAs and mathematical benchmark problems. Comparisons to related published results using the selected circuit examples are carried out, as well as comparisons with the method of directly using EA with exact function evaluations using selected benchmark problems. Concluding remarks are presented in Section V.

II. Motivations and Basic Techniques

A. Challenges for General mm-wave IC Synthesis and Motivations of the GASPAD Approach

General mm-wave IC synthesis involves a medium-scale (15-50 dimensional) computationally expensive constrained optimization problem. A promising solution method is SAEA, which employs surrogate models to replace computationally expensive EM simulations. SAEAs have been applied to mm-wave passive component, IC and antenna synthesis [11], [12], [8]. Its clear advantage in efficiency compared to off-line surrogate model-based methods is discussed in [11]. However, to the best of our knowledge, almost all of these works focus on small-scale problems (i.e., about 5 design variables). Although EMLDE [8] solves medium-scale mm-wave IC synthesis problems, it transforms the problem to a small-scale one, and it is difficult to apply the method for

dimension reduction to general mm-wave IC synthesis, as has been discussed in Section I.

The reason for this "curse of dimensionality" is as follows. A linear increase of the number of design variables leads to an exponential increase of the design space. The newly generated candidate designs in many iterations often spread in various subregions of the design space. To construct reliable surrogate models for prediction or prescreening for the next step of optimization, sufficient training data points around the newly generated candidate designs are necessary, but this is limited by the allowed number of EM simulations (synthesis time). Therefore, it is difficult for a traditional SAEA-based synthesis method to obtain a highly optimized design in an efficient manner for the targeted synthesis problem.

In [10], the authors proposed a novel SAEA for medium-scale computationally expensive optimization, called GPEME. GPEME achieves a comparable solution quality and 2-8 times less expensive exact function evaluations compared to three state-of-the-art SAEAs for medium-scale expensive optimization problems. However, GPEME is for unconstrained optimization. That is another challenge for general mm-wave IC synthesis: when considering both large-signal and small-signal performances, multiple constraints must be handled, some of which may even be tight because of the tough performance specifications. Nevertheless, to the best of our knowledge, most SAEA-based synthesis methods do not handle constraints or use the static penalty function method to deal with constraints. (1) shows the penalized cost function:

$$f^{'}(x) = f(x) + \sum_{i=1}^{i=c} w_i < g_i(x) > \tag{1}$$

where the parameters w_i are the penalty weighting coefficients. $< g_i(x) >$ returns the absolute value of $g_i(x)$ if it is negative, and zero otherwise, considering the constraints $g_i(x) \geq 0, i = 1, 2, \ldots, c$. $f(x)$ is the objective function. Although an SAEA for unconstrained optimization can directly be used when optimizing the penalized function $f^{'}(x)$, the performance of the SAEA will be reduced. The reason is that a continuous and smooth hyper-surface is important for generating high-quality surrogate models, but instead $< g_i(x) >$ are piecewise functions.

Moreover, for the multiple and tight constraints in general mm-wave IC synthesis problems, although there are some state-of-the-art constraint handling methods both in the computational intelligence field [13] and the analog IC sizing field [14], [15], they rely on population updating of a standard EA, and are difficult to be applied to some efficient SAEAs.

Therefore, besides the techniques used in previous SAEA-based synthesis methods, tackling medium-scale problems and multiple tight constraints at the same time is the key motivation for GASPAD.

B. Gaussian Process Machine Learning and Prescreening

Gaussian process (GP) machine learning is used as the surrogate modeling method in GASPAD. GP modeling is a theoretically sound and principled method for determining the much smaller number of free model parameters when compared to many other surrogate modeling approaches such as artificial neural networks [16], [17]. It can also provide an estimate of the model uncertainty for each predicted point, which can be interpreted in a very natural way. It will be discussed now.

1) Gaussian Process Modeling: To model an unknown function $y = f(x), x \in R^d$, GP modeling assumes that $f(x)$ at any point x is a Gaussian random variable $N(\mu, \sigma^2)$, where μ and σ are two constants independent of x. For any x, $f(x)$ is a sample of $\mu + \epsilon(x)$, where $\epsilon(x) \sim N(0, \sigma^2)$. For any $x, x' \in R^d$, $c(x, x')$, the correlation between $\epsilon(x)$ and $\epsilon(x')$, depends on $x - x'$. More precisely,

$$c(x, x') = \exp(-\sum_{i=1}^{d} \theta_i |x_i - x_i'|^{p_i}), \tag{2}$$

where parameter $1 \leq p_i \leq 2$ is related to the smoothness of $f(x)$ with respect to x_i, and parameter $\theta_i > 0$ indicates the importance of x_i on $f(x)$. More details about GP modeling can be found in [18].

2) Hyper Parameter Estimation: Given K points $x^1, \ldots, x^K \in R^d$ and their f-function values $y^1, \ldots, y^K$, then the hyper parameters μ, σ, $\theta_1, \ldots, \theta_d$, and $p_1, \ldots, p_d$ can be estimated by maximizing the likelihood that $f(x) = y^i$ at $x = x^i$ $(i = 1, \ldots, K)$ [19]:

$$\frac{1}{(2\pi\sigma^2)^{K/2}\sqrt{det(C)}} \exp\left[-\frac{(y - \mu 1)^T C^{-1}(y - \mu 1)}{2\sigma^2}\right] \tag{3}$$

where C is a $K \times K$ matrix whose (i, j)-element is $c(x^i, x^j)$, $y = (y^1, \ldots, y^K)^T$ and $\mathbf{1}$ is a K-dimensional column vector of ones.

To maximize (3), the values of μ and σ^2 must be:

$$\hat{\mu} = \frac{\mathbf{1}^T C^{-1} y}{\mathbf{1}^T C^{-1} \mathbf{1}} \tag{4}$$

and

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

Substituting (4) and (5) into (3) eliminates the unknown parameters μ and σ from (3). As a result, the likelihood function depends only on θ_i and p_i for $i = 1, \ldots, d$. (3) can then be maximized to obtain estimates of $\hat{\theta}_i$ and $\hat{p}_i$. The estimates $\hat{\mu}$ and $\hat{\sigma}^2$ can then readily be obtained from (4) and (5).

3) The Best Linear Unbiased Prediction and Predictive Distribution: Given the hyper parameter estimates $\hat{\theta}_i$, $\hat{p}_i$, $\hat{\mu}$ and $\hat{\sigma}^2$, one can predict $y = f(x)$ at any untested point x based on the f-function values y^i at x^i for $i = 1, \ldots, K$. The best linear unbiased predictor of $f(x)$ is [19], [20]:

$$\hat{f}(x) = \hat{\mu} + r^T C^{-1}(y - \mathbf{1}\hat{\mu}) \tag{6}$$

and its mean squared error is:

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

where $r = (c(x, x^1), \ldots, c(x, x^K))^T$. $N(\hat{f}(x), s^2(x))$ can be regarded as a predictive distribution for $f(x)$ given the function values y^i at x^i for $i = 1, \ldots, K$.

4) Blind Gaussian Process Modeling: The surrogate modeling mechanism introduced above is called the ordinary (standard) GP modeling. In the GP modeling of GASPAD, the linear combination of some basis functions $\sum_{i=1}^{m} \beta_i b_i(x)$ is used to replace $\hat{\mu}$ to capture a portion of the variations, which is desirable to represent the general trend of $f(x)$. The goal is to alleviate the complexity of the standard GP modeling when the number of design variables is large, so as to improve the estimation accuracy. This mechanism is called blind GP modeling [21], [22]. For simplicity and efficiency, only linear, quadratic items and two factor interactions are considered as candidate features ($b_i(x)$) in this implementation.

The GP modeling in GASPAD undergoes the following procedures: (1) Based on the available training data points, a standard GP model is firstly constructed, obtaining the estimated hyper parameters. (2) With these hyper parameters and the candidate features, the $b_i(x)$ are ranked based on the estimated $\beta_i (i = 1, \ldots, m)$. The ranking follows a Bayes variable ranking method [21], [22]. (3) The most promising features among $b_i(x) (i = 1, \ldots, m)$ are selected and an intermediate GP model with the original hyper parameters is constructed. Its accuracy is then evaluated by a leave-one-out cross-validation method [21]. This step is repeated until no accuracy improvement can be made. (4) With the selected features and their coefficients, re-optimize the likelihood function and obtain the final GP model.

5) Lower Confidence Bound: We consider the minimization of $f(x)$ in this paper. Given the predictive distribution $N(\hat{f}(x), s^2(x))$ for $f(x)$, a lower confidence bound (LCB) prescreening of $f(x)$ can be defined as [23]:

$$\begin{aligned} &f_{lcb}(x) = \hat{f}(x) - \omega s(x) \\ &\omega \in [0, 3] \end{aligned} \tag{8}$$

where ω is a constant. The use of LCB prescreening can conduct explorative global search when using a large ω or conduct fast local search when using a small ω. Since $\hat{f}(x)$ is Gaussian distributed, according to the 3σ rule, when $\omega = 2$, the confidence level of $f_{lcb}(x)$ to be the LCB of $\hat{f}(x)$ is about 97%. The comparisons of using different ω to define LCB are detailed in [23], [24].

In GASPAD, we use $f_{lcb}(x)$ instead of $\hat{f}(x)$ itself to measure the quality of x for the performance serving as the objective function. For the performances serving as constraints, we use the predicted value $\hat{f}(x)$ itself or $\omega = 0$. It is important to promote explorative global search (by promoting the evaluation of promising but less explored areas, i.e., with high $s(x)$ values) when minimizing the objective function, especially when the objective function is multimodal. On the other hand, we do not expect that many expensive evaluations are spent on nearly (but not) feasible solutions with good objective function values. It is true that such solutions need protection to maintain the diversity in constraint satisfaction, but the prediction uncertainty of the constraint functions provides a natural help. To that end, ω is set to 0 for constraint functions.

C. Differential Evolution

The differential evolution (DE) algorithm is used as the search engine in our proposed GASPAD algorithm. DE is an effective and popular global optimization algorithm. It uses a differential operator to create new candidate solutions [25]. There are quite a few different DE variants. In this paper, we use DE/best/1 to generate new solutions for prescreening. The DE/best/1 mutation uses the current best solution as the base vector, so as to increase the speed of generating promising candidates.

Suppose that P is a population and the best individual in P is x^{best}. Let $x = (x_1, \ldots, x_d) \in R^d$ be an individual solution in P. To generate a child solution $u = (u_1, \ldots, u_d)$ for x, DE/best/1 works as follows.

A donor vector is first produced by mutation:

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

where x^{r_1} and x^{r_2} are two different solutions randomly selected from P and also different from x^{best}. $F \in (0, 2]$ is a control parameter, often called the scaling factor [25]. Then the following crossover operator is applied to produce the child u:

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

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

$$u_j = \begin{cases} v_j, & \text{if } (rand \leq CR) | j = j_{rand} \\ x_j, & \text{otherwise} \end{cases} \tag{10}$$

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

The DE algorithm is shown to be very powerful for real parameter optimization problems. For integer parameters (e.g., the number of fingers of transistors), a quantization method needs to be used [25], which is the same as that in EMLDE [8].

III. The GASPAD Method

A. Active and Passive Components in Synthesis

The parasitic-aware transistor library in this paper is borrowed from existing low-GHz RF IC synthesis methods [3] and is the same as that in EMLDE [8]. The minimum transistor length and a fixed transistor width are used, while only the number of fingers is changed. The transistor layouts with different number of fingers are prepared before the synthesis and the parasitics of each of them are extracted and saved for future full-fletched simulations.

At mm-wave frequencies, it is a general routine to employ integrated transformers for impedance matching between transistors to reduce the insertion loss and to ensure a compact layout [26]. However, only using conventional transformers may not obtain high performance in many occasions [27]. Although the impedance matching conditions can be improved by carefully tuning the width and diameter of the primary and secondary windings, the power transfer efficiency of the transformer may then be compromised. To address this problem, a solution is to add transmission lines at the input and output of the transformer. However, co-optimization of each

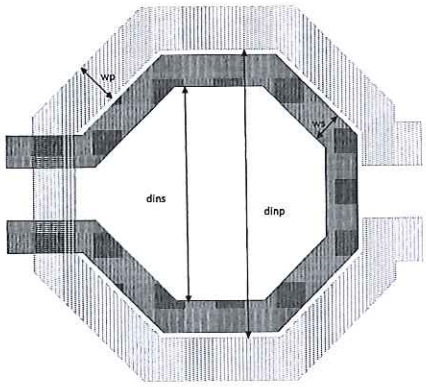


Fig. 1. The conventional transformer matching network (4 parameters)

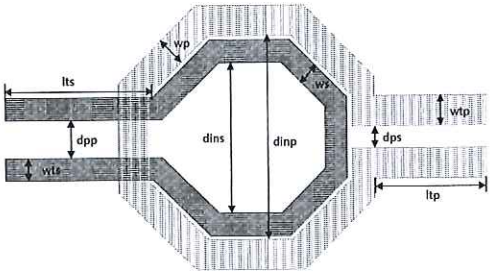


Fig. 2. The transformer with transmission line matching network (10 parameters)

transformer and its transmission lines is not straightforward for designers. Quite a number of iterations are required to finalize such matching network and the performance may still not be good enough.

In our proposed GASPAD method, the transformer with transmission lines at its primary and secondary ports is considered as a single component in synthesis. Clearly, the tedious tuning of the design parameters of this matching network based on "experience and trial" is not needed anymore, but the cost to pay is the higher dimensionality of the synthesis problem. In other words, we shift the "headache" of manual design to computational intelligence algorithms, which requires to develop powerful SAEAs for medium-scale expensive optimization problems as presented in this paper.

The conventional transformer matching network (XFMR) and the transformer with transmission line matching network (XFMRTL) and their design parameters are shown in Fig. 1 and Fig. 2, respectively. Fig. 1 shows the matching network using the conventional transformer. The design parameters are the inner diameters of the primary and secondary inductors ($dins$, $dinp$) and the metal width of these two inductors (ws, wp). Fig. 2 shows the matching network using the transformer with extra transmission lines. Besides the four design parameters used in the above transformer, the four additional design parameters are the length of the transmission lines (ltp, lts) and the widths of the transmission lines (wtp, wts). In addition, the distances between the two ports (dpp, dps) on each side are two new design parameters to enhance the flexibility, while they are often fixed by designers in the XFMR matching network.

B. Main Ideas of the SAEA Framework in GASPAD

The SAEA-based synthesis methods for small-scale problems [8], [11] select in each iteration the best generated candidate design based on prescreening for full-fletched simulation, and then update the population. GASPAD inherits the selection of the current best candidate design for simulation, but three substantial algorithmic developments are introduced, which are the integration of the constraint handling method, the surrogate model-aware evolutionary search mechanism and the composition of the training data points.

GASPAD integrates constraint handling into the rules for ranking the newly generated candidates. In other words, constraint satisfaction is considered to define the "best" candidate design in each iteration. The following ranking rules are presented:

1: The feasible design solutions (if any) rank higher than the infeasible design solutions.

2: The feasible design solutions (if any) are ranked based on the sorting of the objective function values in ascending order (considering a minimization problem).

3: The infeasible design solutions are ranked based on the sorting of the sum of the constraint violation values in ascending order.

It can be seen that the ranking rules use the basic idea of a tournament selection method for constrained optimization [28], which is widely used in the evolutionary computation (EC) field. Nevertheless, tournament selection based on a standard EA population is not used, but is revised to focus on the current best candidate design in order to match the proposed SAEA. Assuming that the prescreened best candidate design is a top ranked one in the generated candidate designs, the evolution can be divided into three phases. From the beginning to the appearance of the first feasible solution, GASPAD aims at minimizing the constraint violations (e.g., satisfying the G_p, PAE specifications). From the appearance of the first feasible solution to where considerable number of solutions are feasible in the current parent population, GASPAD searches for both objective function optimization (e.g., optimizing P_{1dB}) and constraint satisfaction. After that till the end of the synthesis, GASPAD concentrates on optimizing the objective function. Note that independent surrogate models are constructed for each constraint, and this does not affect the smoothness and continuity of the hyper-surface of the objective function and the constraint functions.

The challenge for SAEA to handle medium-scale synthesis problems is that the training data (i.e., the evaluated candidate solutions) for promising subregions may not be sufficient. This is a contradiction in SAEA, and is especially obvious when the design space is large, since EA searches the design space for the sake of optimization, and overlooks the requirement of producing high-quality surrogate models. The star points in Fig. 3 show a typical spreading of the training data pool in

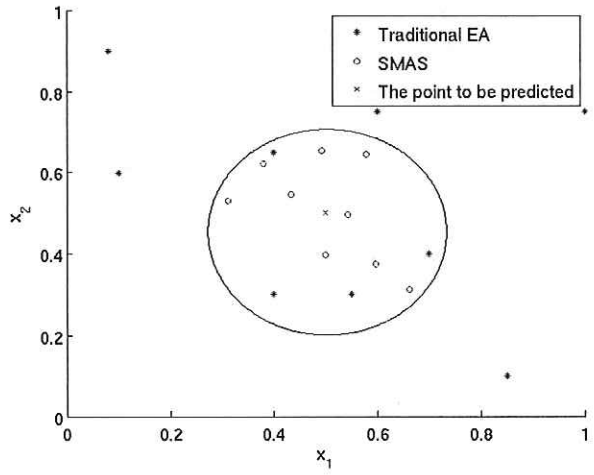


Fig. 3. An illustrative figure for the locations of the training data pool by different search mechanisms

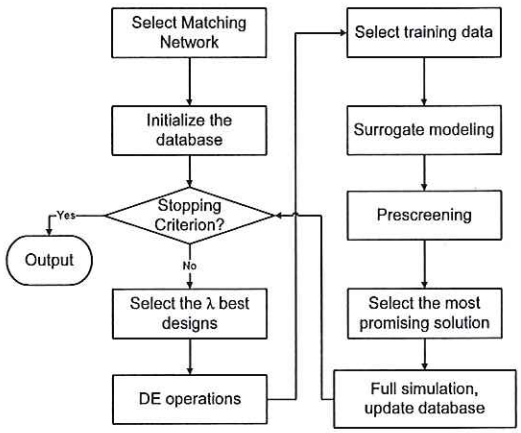


Fig. 4. The flow diagram of GASPAD

two of the d dimensions when using standard EA operators and population updating. The current promising subregion is shown by the ellipse and a point waiting to be predicted or prescreened is shown by the cross. It can be seen that the already evaluated candidate solutions spread in different search subregions. When using the whole training data pool, the points far away from the point with the cross will, on the contrary, deteriorate the quality of the constructed surrogate model. Note that this is unlike off-line surrogate modeling, whose training data points are intentionally located almost uniformly. On the other hand, there are not enough training data points in the current promising area to produce a high-quality surrogate model.

Given that the total number of computationally expensive evaluations (EM simulations) is restricted by the synthesis time and standard EA operators need to be used, our idea is to develop a new population updating mechanism in order to improve the locations of the training data pool (i.e., evaluated candidate solutions) to help surrogate modeling while keeping a good optimization ability. We call it surrogate model-aware evolutionary search (SMAS) mechanism. There are two elementary factors to achieve this goal: (1) the search should be concentrated on the current promising area for helping surrogate modeling and exploitation at the same time; (2) a reasonable diversity should be maintained to guarantee the exploration ability. The expected locations of the training data pool produced by our proposed search mechanism are shown by circles in Fig. 3.

Instead of using a standard EA population [24] or using a continuously increasing population [11] in SAEA, the λ current best candidate designs form the parent population (it is reasonable to assume that the search focuses on the promising subregion) and the best candidate based on prescreening in the child population is selected to replace the worst one in the parent population at each iteration. In this way, only at most one candidate is changed in the parent population at each iteration, so the best candidate in the child solutions in several consecutive iterations may be quite near (they will then be simulated and are used as training data). Therefore, the training data describing the current promising region can be much denser compared to those generated by a standard EA population updating. To maintain the exploration ability, the whole child population is generated by the DE mutation and crossover operators at each iteration. Our pilot experiments on more than 10 complex mathematical benchmark problems in the EC field show good convergence of SMAS.

Appropriate training data points need to be selected to describe the current promising area considering both the SMAS mechanism and the constraint satisfaction. In GASPAD, the nearest τ training data from the median of candidates in the child population waiting to be prescreened / predicted is used. A single surrogate model is constructed based on them and is used to prescreen / predict the whole child population. It is straightforward that the value of τ is in proportion to the number of design variables d. This will further be discussed in Section III (E).

C. The General Framework of the GASPAD Algorithm

GASPAD records all the evaluated solutions and their function values in a database. Once a full-fletched simulation has been conducted for a new candidate design x, the x and its performances y will be added to the database. To initialize the database, a Design of Experiments method, Latin Hypercube sampling (LHS) [29], is used to sample a set of initial points from the search space. This step is the same as in EMLDE. For details, please see [8].

The flow diagram of the GASPAD method is shown in Fig. 4. It works as follows:

Note that the selected candidate solution is not necessarily the real best one in terms of exact simulation; it is satisfactory that the prescreened best one is among the top few best candidates in reality.

Step 1: Select the form of the matching network used (XFMR, XFMRTL). Decide the design space $[a, b]^d$, where a and b are the lower and upper bounds of the design variables, respectively.

Step 2: Use LHS to sample α candidate designs from $[a, b]^d$. Perform full-fletched simulations (EM simulations, circuit simulations) to all of these candidate designs and let them form the initial database.

Step 3: If a preset stopping criterion is met (e.g., a threshold of synthesis time, a certain number of iterations), output the best design in the database; otherwise go to step 4.

Step 4: Select from the database the λ best candidate designs in terms of simulation results according to the ranking rules described in Section III (B) to form a population P.

Step 5: Apply the DE operators ((9) and (10)) on P to generate λ child solutions.

Step 6: Calculate the median of the λ child solutions to obtain the vector mx. Take the τ nearest solutions to mx in the database (based on Euclidean distance) and their function values (performances) as the training data to construct GP surrogate models.

Step 7: Prescreen the λ child solutions generated in Step 5 by using the GP model with LCB prescreening for the objective function and the predicted values for each constraint.

Step 8: Perform full-fletched simulation to the estimated best child solution from Step 7. Add this evaluated design and its performances to the database. Go back to Step 3.

D. Discussions

1) Effectiveness of GASPAD for expensive constrained optimization: GASPAD uses the proposed ranking rules with the SMAS mechanism to address expensive constrained optimization problems. One may have two questions: (1) Assuming surrogate models with good quality can be constructed, can this new search mechanism solve constrained optimization problems effectively? (2) Can the surrogate modeling really work as expected in this framework? Experiments using 10 tightly constrained mathematical benchmark problems [13] have been carried out and highly optimized results have been obtained. In the following, we use an example (see P_1 in the appendix) to clarify the above two questions.

For the first question, we simulate GASPAD. Step 6 to Step 8 of GASPAD (see Section III (C)) construct the surrogate model and select the prescreening-based best child solution. To simulate it, we remove the GP modeling and prescreening. Instead, we conduct exact function evaluations to all the λ child solutions in each iteration, and randomly select one from the top γ solutions. γ is set to 5 and 10 runs are performed. The parameter settings are detailed in Section IV. All the results are feasible and the worst result is 25.11, compared to the globally optimum 24.31. Hence, the high search ability of GASPAD is shown.

For the second question, we conduct exact function evaluations to all the child solutions in each run of GASPAD (not the simulated GASPAD as above) and calculate the real ranks of the prescreened best candidates in all the iterations. 10 runs are performed. The results are similar to the simulated results and the prescreened best candidate is among the top 3 and top 5 of the child population in 86.1% and 91.1% of the iterations, respectively. It can be seen that the surrogate modeling and prescreening work as expected.

The weakness of using penalty functions together with unconstrained expensive optimization methods was discussed in Section II (A). A verification is provided here. The same SMAS mechanism (including parameters) for unconstrained expensive optimization is applied with a penalized function of P_1. To address the well-known problem of inappropriate penalty coefficients, we use a state-of-the-art method, self-adaptive penalty function (SAPF) [30]. We call this method SAPF-SMAS. 10 runs for simulated SAPF-SMAS and 10 runs for real SAPF-SMAS are performed. The simulated SAPF-SMAS shows similar results compared to GASPAD. However, the real SAPF-SMAS shows worse results and robustness compared to GASPAD. All the results are worse than the worst result of GASPAD and its worst result is 32.66. This shows that the surrogate modeling and prescreening do not work as good as expected. The prescreened best candidate is among the top 3 and top 5 of the child population in 20.6% and 32.2% of the iterations, respectively, in contrast to that of GASPAD.

2) The advantage of SMAS: The method to integrate surrogate modeling into EA is the key of SAEA research. The search ability (including optimality and diversity) and the surrogate model quality need to be considered simultaneously in order to optimally trade-off the efficiency (number of simulations) and the effectiveness (solution quality). Many available methods in the computational intelligence field and most methods in the EDA field utilize the individual / generation control-based technique [31], such as [9], [32], [33], [34]. This kind of methods holds the basic idea of using the standard optimization method to tackle optimality and diversity and using surrogate model(s) as the major tool only when its prediction uncertainty is small. Good optimization quality can thus be maintained. Also owing to this, such methods consume more expensive evaluations than necessary, since almost all the subregions visited by the search operators are carefully modeled, but many of them are far from optimal. In contrast, SMAS replaces the standard EA population updating by a new mechanism with the goal of effectively supporting surrogate modeling with highly reduced needed number of simulations. At the same time, it maintains the optimality and diversity for effective search.

Experimental verifications for more than 10 complex benchmark problems show the high efficiency and effectiveness of SMAS. Here we use an example (see P_3 in the Appendix, which is a 30-dimensional highly multimodal problem with global optimum 0) and compare SMAS with a state-of-the-art generation control-based method with adaptive local search and modeling, GS-SOMA [9]. Because there are few reported SAEA results for expensive constrained optimization problems, we use unconstrained problems and compare the results of the SMAS mechanism in GASPAD to published results of GS-SOMA for P_3 . The median of the optimal value

using only 1000 exact function evaluations by SMAS over 20 runs is 1.67 (the worst result is 2.83), while the median result of GS-SOMA using 8000 evaluations is 3.67. (In [9], 8000 evaluations are used).

E. Parameter Settings

There are several control parameters in GASPAD. Besides some parameters that have been well studied in literature (e.g. the DE parameters, the LCB parameter), the recommended setting rules of other parameters are as follows. They are empirical rules based on tens of constrained and unconstrained mathematical benchmark problems with different properties and scales.

- The scaling factor F and the crossover rate CR in the DE operators: The setting of the DE parameters has been well investigated. Following [25], we set F to 0.8, and CR to 0.8. The setting of $CR = 0.8$ works well for most problems. For some very complex or ill-defined problems, CR can be decreased.
- The number of training data points τ in the GP modelling: τ is suggested to be between $5 \times d$ to $7 \times d$, where d is the number of design variables.
- ω used in LCB: Following the suggestions in [23], [24], $\omega = 2$ is used.
- The number of initial samples α: Our empirical rule is that α should be set to at least $3 \times d$ or the robustness will decrease. α is affected by the complexity of the function. For highly multimodal problems, $\alpha = 5 \times d$ is often enough. We can set α between 60 and 200 for a 15-50 dimensional mm-wave IC synthesis problem.
- The population size λ: This is a DE parameter. Although GASPAD has a completely different population updating method compared to standard DE, our pilot experiments showed that the recommended setting of DE population size [25] is still applicable. Using $30 \leq \lambda \leq 60$ often works well. A large λ value causes a slow convergence and a small value can easily lead to premature convergence.

IV. Experimental Results and Comparisons

PA design is selected as the example of mm-wave IC design in this section. PA design is very difficult in the mm-wave IC design area, because there are tedious tuning iterations between load-pull simulations and the design of impedance matching networks [26], [27]. Moreover, at mm-wave frequencies, not only the output matching network, but also the input and inter-stage matching networks need to be optimized to ensure sufficient output power and a high efficiency.

In this section, the GASPAD method is first demonstrated for the synthesis of two 60GHz PAs in a 65nm CMOS technology. ADS-Momentum is used as the EM simulator. Cadence SpectreRF is used as the circuit simulator. The bounds of the design variables are set both by the design rules of the technology and the experience of the designer. The two examples are all constrained optimization problems, with constraints on the 1dB compression point (P_{1dB}) and the power gain (G_p), and the objective function is the power added efficiency (PAE) at P_{1dB}. GASPAD stops when the performance cannot be improved for 50 consecutive iterations. The examples are run on an Intel 2.66GHz dual Xeon PC under the Linux operating system and the MATLAB environment. All the time consumptions mentioned in the experiments are wall clock time.

The performance of an SAEA may be affected by the random numbers used in the evolution operators. To test the robustness of GASPAD and to compare GASPAD with an EA-based constrained optimization method with exact function evaluations, mathematical benchmark problems are used. Benchmark problems in the EC field are often complex problems with different properties, such as having many locally optimal points, being discontinuous and having tight constraints. [8] has demonstrated the use of proper benchmark problems to test RF IC synthesis methods. Two benchmark problems are used and 20 runs with independent random numbers are performed for each of them.

The parameter setting rules for the GASPAD method are demonstrated in Section III (E). Besides the parameters with a single fixed recommended value (e.g., the DE mutation rate), as described above, the number of training data points (τ) is set to $5 \times d$ and the population size (λ) is set to 40 for all the experiments in this section. For the number of initial samples (α), α is set to 120 when the number of design variables is larger than 30, and α is set to 70 when the number of design variables is less than or equal to 20. For the GP modeling, the ooDACE toolbox [22] is used. Note that the design variables and performances need to be normalized. GP machine learning assumes that the inputs and outputs are samples of a Gaussian distributed variable. In our implementation, they are scaled such that each variable is distributed with a mean of 0 and a variance of 1.

A. Test Example 1

The first example is a 60GHz two-stage PA with cascode differential pairs. The transistor in the output stage has six fingers and the number of fingers of a transistor (nf) in the driver stage is a design variable. The layouts of the transistors with different number of fingers are designed beforehand by the designer. The output load impedance is 50Ω. The schematic is shown in Fig. 5. Because the XFMR matching network is used, the design variables for the passive components are $dins$, $dinp$, ws and wp for each of the three transformers. (For details, see Section III (A).) There are 5 biasing voltages: V_{DD}, V_{cas1}, V_{cas2}, V_{b1} and V_{b2}. The ranges for the design variables are summarized in Table I. There are in total 18 design parameters.

The synthesis problem is:

$$\begin{array}{ll} \text{maximize} & PAE(@P_{1dB}) \\ \text{s.t.} & P_{1dB} \geq 13\text{dBm} \\ & G_p \geq 10\text{dB} \end{array} \tag{11}$$

After 204 evaluations, GASPAD obtained the optimized design. The layout of the synthesized PA is shown in Fig. 6. The 1dB compression point is 14.87 dBm, the power added

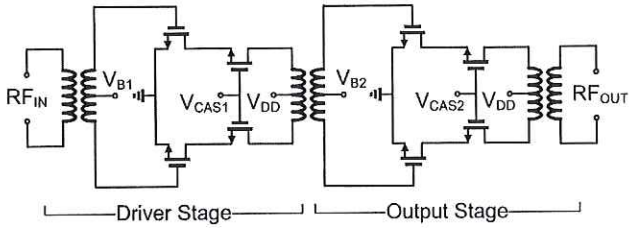


Fig. 5. Schematic of the 60GHz power amplifier (example 1)

TABLE I
DESIGN PARAMETERS AND THEIR RANGES FOR THE 60GHZ POWER AMPLIFIER (EXAMPLE 1)

Parameters	Lower Bound	Upper Bound
$dinp, dins(\mu m)$	20	100
$wp, ws(\mu m)$	3	10
$V_{DD}(V)$	1.5	2
$V_{cas1}(V)$	1.2	2
$V_{cas2}(V)$	1.2	2
$V_{b1}(V)$	0.55	0.95
$V_{b2}(V)$	0.55	0.95
nf (integer)	2	5

efficiency at P_{1dB} is 9.85% and the power gain is 10.73 dB. S-parameter simulation shows that the lowest Rollet stability factors (K factors) is 10.68, which is larger than 1, and $|\Delta|$ is smaller than 1, so the obtained circuit design is unconditionally stable [35]. The simulation results are shown in Fig. 7. The time consumption for GASPAD to synthesize this PA is 42 hours.

B. Test Example 2

The second example is a 60GHz two-stage PA, and each stage is realized by a neutralized common-source (CS) [26] stage. The transistor in the output stage has six fingers and the number of fingers of a transistor in the driver stage is a design variable. Also, the layouts of the transistors with different number of fingers are designed beforehand by the designer. The output load impedance is 50Ω. The schematic is shown in Fig. 8. Two experiments are carried out. The first one uses the XFMR matching network, while the second one uses the XFMRTL matching network. There are two biasing voltages: V_{b1} and V_{b2}. The supply voltage of this example is set to 1V. The ranges for the design variables are summarized in Table II. For the PA using the XFMR matching network,

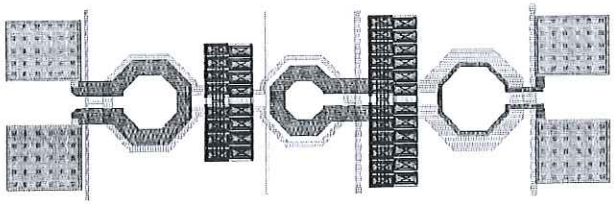
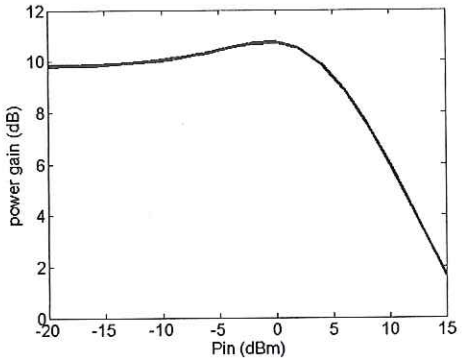
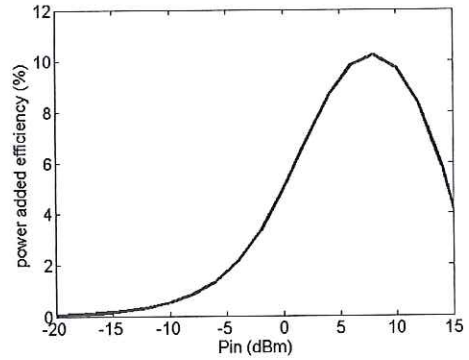


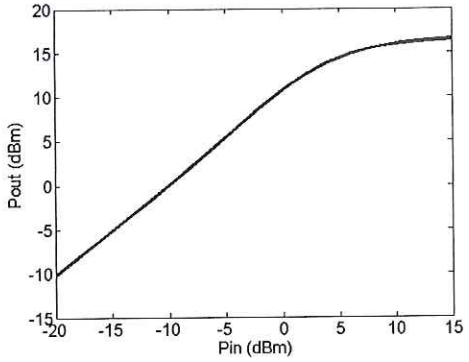
Fig. 6. Layout of the PA synthesized by GASPAD (example 1)



(a) Simulated power gain of the 60GHz PA (example 1)



(b) Simulated power added efficiency of the 60GHz PA (example 1)



(c) Simulated output power of the 60GHz PA (example 1)

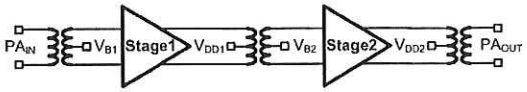
Fig. 7. The simulated performances of the 60GHz PA (example 1) synthesized by GASPAD

there are 15 parameters, while for the PA using the XFMRTL matching network, there are 33 design parameters.

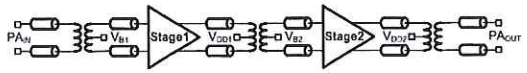
The synthesis problem is:

$$\begin{array}{ll} \text{maximize} & PAE(@P_{1dB}) \\ \text{s.t.} & P_{1dB} \geq 10\text{dBm} \\ & G_p \geq 10\text{dB} \end{array} \tag{12}$$

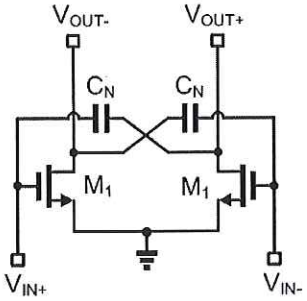
The PA using the XFMR matching network is synthesized first. After 427 evaluations, GASPAD obtained the optimized design. The layout of the synthesized PA is shown in Fig. 9. The 1dB compression point is 10.12dBm, the power added efficiency at P_{1dB} is 10.14% and the power gain is 16.22dB.



(a) 60GHz PA using XFMR matching network (example 2a, 15 parameters)



(b) 60GHz PA using XFMRTL matching network (example 2b, 33 parameters)



(c) neutralized common-source (CS) amplifier stage (used in example 2)

Fig. 8. Schematic of the 60GHz power amplifier (example 2)

TABLE II
DESIGN PARAMETERS AND THEIR RANGES FOR THE 60GHZ POWER AMPLIFIER (EXAMPLE 2)

Parameters	Lower Bound	Upper Bound
$dinp, dins(\mu m)$	20	100
$wp, ws(\mu m)$	3	10
$ltp, lts(\mu m)$	0.1	50
$wtp, wts(\mu m)$	3	10
$dpp, dps(\mu m)$	3	10
$V_{b1}(V)$	0.55	0.95
$V_{b2}(V)$	0.55	0.95
nf (integer)	2	5

The time consumption for GASPAD to synthesize this PA is 76 hours. In the synthesis, we found that after 224 iterations (about 44 hours), there is a solution satisfying the constraints on P_{1dB} and G_p, with a $PAE(@P_{1dB})$ value of 9.02%. After that, the $PAE(@P_{1dB})$ value improves gradually to 10.14%. Another experiment is done for this circuit, where by setting the P_{1dB} constraint to be larger than 11dBm, the final $PAE(@P_{1dB})$ value is decreased to less than 10%. It can be seen that when using the XFMR matching network, there is a trade-off between $PAE(@P_{1dB})$ and P_{1dB} when they are near 10% and 10dBm, respectively.

To enhance the performance, we then use the XFMRTL matching network with the same specifications and optimization objective of (12). After 581 evaluations, GASPAD obtained the optimized design with P_{1dB}=11.47dBm, G_p=11.23dB and $PAE(@P_{1dB})$=13.58%. S-parameter simulation shows that the lowest K factor is 24.55, which is larger than 1, and $|\Delta|$ is smaller than 1, so the obtained circuit design

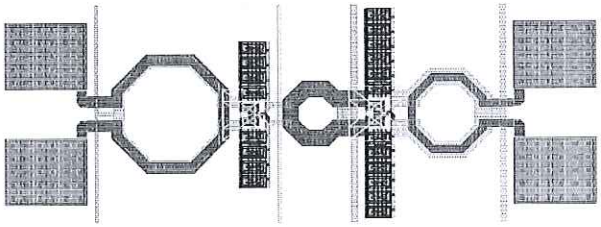


Fig. 9. Layout of the PA synthesized by GASPAD (example 2a, 15 parameters)

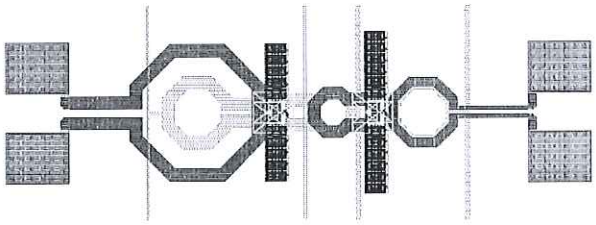


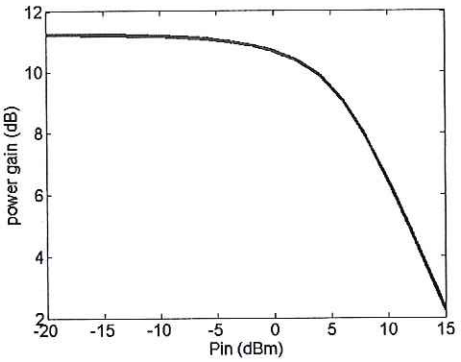
Fig. 10. Layout of the PA synthesized by GASPAD (example 2b, 33 parameters)

is unconditionally stable [35]. The layout of the synthesized PA is shown in Fig. 10. The simulation results are shown in Fig. 11. The time consumption of GASPAD to synthesize this PA is 106 hours. Clearly, the XFMRTL matching network has improved the performances and GASPAD has successfully solved this 33-parameter synthesis problem.

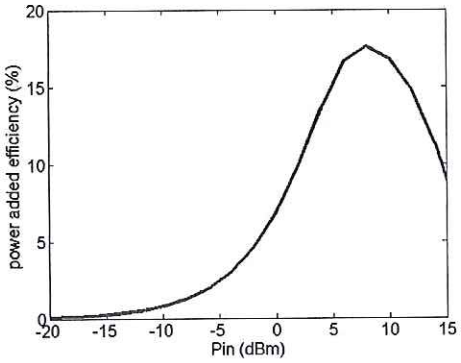
C. Comparisons with the State-of-the-art

Since a single simulation of a typical PA design costs 10 to 13 minutes, comparing with the conventional constrained optimization algorithms costs a too long time. For example, for the 581 evaluations used in the 33-parameter synthesis problem of example 2, it is about 15 iterations for standard DE when the population size (λ) is 40. For typical medium-scale constrained optimization problems, hundreds to thousands of iterations are often needed based on a standard EA to get to a similar result. Hence, we refer to the state-of-the-art 65nm PA performances in literature. Although it is not fully precise to compare post-simulation results with measurement results, the quality of the design solutions obtained by GASPAD can nevertheless be observed.

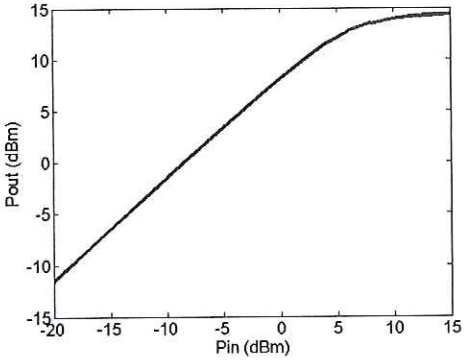
Table III shows the performances of five state-of-the-art PA designs in 65nm CMOS technologies. The reported 60GHz PAs are mainly optimized for saturated output power P_{sat} and maximum power added efficiency ($PAE(@P_{sat})$). However, as was said in Section I, optimizing $PAE@P_{1dB}$ is more essential, but is difficult to achieve by the available manual design methods. Some $PAE@P_{1dB}$ values of these circuits are estimated from the figures in these papers. Note also that the power combining technique is used in some of these designs, which combines the output power of several PAs and allows to improve the P_{1dB} performance, whereas this technique is not used in the examples in this paper. Some referred PAs also have three stages, which leads to a higher



(a) Simulated power gain of the 60GHz PA (example 2b, 33 parameters)



(b) Simulated power added efficiency of the 60GHz PA (example 2b, 33 parameters)



(c) Simulated output power of the 60GHz PA (example 2b, 33 parameters)

Fig. 11. The simulated performances of the 60GHz PA (example 2b, 33 parameters) synthesized by GASPAD

power gain.

From Table III, it is clear that the designs synthesized by GASPAD are of very high quality.

It is interesting to discuss the relation of EMLDE [8] and GASPAD. Compared to EMLDE, GASPAD possesses a much stronger surrogate model assisted search method. On the other hand, a main novelty in EMLDE is the decomposition method which transforms a medium-scale synthesis problem into a small-scale one, which is much easier to solve. For problems that are capable for EMLDE (i.e., mm-wave IC synthesis

TABLE III
60GHz 65nm PAs Performances

references	$P_{1dB}(dBm)$	PAE $(@P_{1dB})$ (%)	$G_p(dB)$	topology
[27]	15.0	6.8	20.3	3-stage CS, 2-way power combining
[36]	15.1	6.5	19.2	3-stage, 4-way power combing
[26]	5.0	5.0	16.0	3-stage neutralized CS
[37]	6.8	7.7	30.0	3-stage neutralized CS
[38]	11.5	2.0	15.5	2-stage cascode, 8-way power combining
example 1	14.87	9.85	10.73	2-stage cascode
example 2(a)	10.12	10.14	16.22	2-stage neutralized CS
example 2(b)	11.47	13.58	11.23	2-stage neutralized CS

only considering small-signal performances), directly applying GASPAD will have a reduced efficiency, although still practical. For those problems, a suggested method is to apply GASPAD to each sub-problem decomposed by EMLDE.

D. Benchmark Problem Tests

[8] suggests to use proper mathematical benchmark problems in the EC field to test RF IC synthesis methods. This has three advantages: (1) fast exact function evaluations for having sufficient runs to test the robustness and to compare with other methods; (2) not limited to the technology used; (3) with known global optimal solution, providing a criterion to judge the quality of a method. For the general mm-wave IC synthesis problems, three factors are necessary to be considered, which are: medium-scale, tight constraints (high-performance specifications) and multimodal (local optima). However, most of the available benchmark problems for tightly constrained optimization problems are small-scale problems (about 5 variables). Thus, we select a tightly constrained optimization problem with 10 variables (P_1) and a self-constructed problem (P_2) based on benchmark problems for unconstrained optimization with 20 variables. The latter problem is highly multimodal (numerous locally optimal points) both for the objective function and the constraints. The two test problems (P_1 and P_2) are in the Appendix.

The GASPAD method is tested by these two problems using 20 runs. The parameters are the same as those in Section IV (A). 1000 exact function evaluations are used for P_1 and 1500 exact function evaluations are used for P_2. The results are shown in Table IV.

From Table IV, it can be seen that the GASPAD method has a good performance for the tested benchmark problems. For P_1, although the worst result is 107.82, such failure run only happened a single time and the second worst result is 27.9,

TABLE IV
STATISTICS OF THE BEST FUNCTION VALUES OBTAINED BY GASPAD OVER 20 RUNS

Problem	Opt.	best	worst	median	R_{inf}
P_1	24.31	24.31	107.82	24.32	0
P_2	0	3.57e-9	0.033	5.49e-9	0

Opt. is the globally optimal objective function value of each problem. R_{inf} refers to the number of runs whose final result is infeasible.

TABLE V
STATISTICS OF THE BEST FUNCTION VALUES OBTAINED BY SBDE OVER 20 RUNS

Problem	Opt.	best	worst	median	R_{inf}
P_1	24.31	24.31	512.25	24.31	0
P_2	0	0	0.02	0.015	0

Opt. is the globally optimal objective function value of each problem. R_{inf} refers to the number of runs whose final result is infeasible.

compared to the globally optimal solution 24.31. For P_2, all the results are very good.

The efficient tournament selection-based constraint handling method [28] is widely used in various fields. Next, we compare GASPAD with the selection-based differential evolution (SBDE) for constrained optimization. SBDE combines the tournament selection-based constraint handling method [28] and the DE algorithm. The details of using this method for RF passive components and for IC synthesis (without surrogate model) are in [8], [11]. The DE parameters and the population size λ are the same as those in Section IV (A) and 20 runs are performed. Note that no surrogate model is used, and the number of evaluations is 40×1000 and 40×1500, respectively. The results are shown in Table V.

Comparing Table V with Table IV, it can be seen that GASPAD and SBDE have obtained comparable results for these two problems in terms of solution quality. Then, we compare the number of exact function evaluations, which dominates the efficiency for computationally expensive optimization problems. Since the convergence happens before using all the allowed function evaluations, to mark the convergence of GASPAD, we use a threshold, δ, which means that after N_{ec} evaluations, the current best solution is feasible and the improvement to the objective function is less than δ after that. Thus, we consider GASPAD converge at N_{ec} evaluations. $\delta = 0.1$ is used and we compare the median of the best objective function values found so far in each iteration for each run by both methods. We take the following information from the available data:

- $G_{N_{ec}}$: the median of the best function values obtained using N_{ec} exact function evaluations by GASPAD;
- $S_{N_{ec}}$: the median of the best function values obtained using N_{ec} exact function evaluations by SBDE;
- $H_{N_{ec}}$: the number of exact function evaluations needed for SBDE to achieve the same function value as GASPAD does.

The comparison results are shown in Table VI.

From the mathematical benchmark problem tests, it can be seen that GASPAD can obtain a comparable result with about 20 times less exact function evaluations than the popular SBDE method.

TABLE VI
COMPARISONS BETWEEN GASPAD AND SBDE OVER 20 RUNS

Problem	N_{ec}	$G_{N_{ec}}$	$S_{N_{ec}}$	$H_{N_{ec}}$	speedup
P_1	884	24.4	1096.2	20280	23
P_2	732	0.097	148.29	12760	17

V. CONCLUSIONS

In this paper, the first generalized mm-wave IC synthesis method, called GASPAD, has been presented. GASPAD achieves the three stated goals (being general for mm-wave IC synthesis; being effective for obtaining highly optimized design solutions; being efficient for finishing the synthesis in a practical and acceptable time). This has been shown by experiments on 60GHz power amplifiers and mathematical benchmark problems. In addition, GASPAD has advantages compared to available "experience and trial"-based manual design methods, and is not difficult to implement. The combined high optimization ability and high efficiency is achieved by the core ideas of the surrogate model-aware evolutionary search mechanism (SMAS) that is suited for tackling tens of design variables, and the extension and integration of a selection-based constraint handling method into SMAS for tackling multiple design specifications. Future works include developing GASPAD-based software tools and developing multi-objective mm-wave IC synthesis methods. Readers can contact liubo168@gmail.com for theory and implementation of GASPAD.

ACKNOWLEDGEMENT

The authors would like to thank PD Dr. Helmut Graeb for valuable discussions.

APPENDIX

A. Benchmark Test Problem P_1

$$
\begin{aligned}
min\quad & f(\vec{x}) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3-10)^2 \\
& +4(x_4-5)^2 + (x_5-3)^2 + 2(x_6-1)^2 + 5x_7^2 \\
& +7(x_8-11)^2 + 2(x_9-10)^2 + (x_{10}-7)^2 + 45 \\
s.t.\quad & g_1(\vec{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \le 0 \\
& g_2(\vec{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \le 0 \\
& g_3(\vec{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \le 0 \\
& g_4(\vec{x}) = 3(x_1-2)^2 + 4(x_2-3)^2 + 2x_3^2 - 7x_4 - 120 \le 0 \\
& g_5(\vec{x}) = 5x_1^2 + 8x_2 + (x_3-6)^2 - 2x_4 - 40 \le 0 \\
& g_6(\vec{x}) = x_1^2 + 2(x_2-2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \le 0 \\
& g_7(\vec{x}) = 0.5(x_1-8)^2 + 2(x_2-4)^2 + 3x_5^2 - x_6 - 30 \le 0 \\
& g_8(\vec{x}) = -3x_1 + 6x_2 + 12(x_9-8)^2 - 7x_{10} \le 0 \\
& x_i \in [-10, 10], i = 1, \ldots, 10 \\
minimum & : f(x^*) = 24.30620906818
\end{aligned}
\tag{13}
$$

B. Benchmark Test Problem P_2

$$
\begin{aligned}
&min \quad f(\vec{x}) = 1 + \sum_{i=1}^{20} \frac{(100*x_i)^2}{4000} - \prod_{i=1}^{20} cos(\frac{(100*x_i)}{\sqrt{i}}) \\
&s.t. \quad g_1(\vec{x}) = -20e^{-0.2\sqrt{\frac{1}{20}\sum_{i=1}^{20} x_i^2}} - e^{\frac{1}{20}\sum_{i=1}^{d} cos(2\pi x_i)} \\
&-5 \leq 0 \\
&\qquad g_2(\vec{x}) = -\sum_{i=1}^{20} x_i - 10 \leq 0 \\
&\qquad x_i \in [-6, 6], i = 1, \ldots, 20 \\
&minimum : f(x^*) = 0
\end{aligned}
\tag{14}
$$

C. Benchmark Test Problem P_3

$$
\begin{aligned}
&min \quad f(x) = -20e^{-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30} x_i^2}} - e^{\frac{1}{30}\sum_{i=1}^{d} cos(2\pi x_i)} \\
&\qquad x \in [-32.768, 32.768], i = 1, \ldots, 30 \\
&minimum : f(x^*) = 0
\end{aligned}
\tag{15}
$$

REFERENCES

[1] A. Niknejad, "Siliconization of 60 GHz," *Microwave Magazine, IEEE*, vol. 11, no. 1, pp. 78–85, 2010.

[2] D. Zhao, S. Kulkarni, and P. Reynaert, "A 60-GHz outphasing transmitter in 40-nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 47, no. 12, pp. 3172–3183, 2012.

[3] D. J. Allstot, J. Park, and K. Choi, *Parasitic-aware optimization of CMOS RF circuits*. Springer, 2003.

[4] G. Tulunay and S. Balkir, "A synthesis tool for CMOS RF low-noise amplifiers," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 27, no. 5, pp. 977–982, 2008.

[5] A. Agarwal and R. Vemuri, "Layout-aware RF circuit synthesis driven by worst case parasitic corners," in *Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings. 2005 IEEE International Conference on*. IEEE, 2005, pp. 444–449.

[6] G. Zhang, A. Dengi, R. A. Rohrer, R. A. Rutenbar, and L. R. Carley, "A synthesis flow toward fast parasitic closure for radio-frequency integrated circuits," in *Proceedings of the 41st annual Design Automation Conference*. ACM, 2004, pp. 155–158.

[7] M. Chu and D. J. Allstot, "Elitist nondominated sorting genetic algorithm based RF IC optimizer," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 52, no. 3, pp. 535–545, 2005.

[8] B. Liu, N. Deferm, D. Zhao, P. Reynaert, and G. Gielen, "An efficient high-frequency linear RF amplifier synthesis method based on evolutionary computation and machine learning techniques," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 31, no. 7, pp. 981–993, 2012.

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

[10] B. Liu, Q. Zhang, and G. Gielen, "A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive black box optimization problems," *Evolutionary Computation, IEEE Transactions on (Accepted)*.

[11] 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.

[12] 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.

[13] R. Mallipeddi and P. Suganthan, "Ensemble of constraint handling techniques," *Evolutionary Computation, IEEE Transactions on*, vol. 14, no. 4, pp. 561–579, 2010.

[14] F. Medeiro, R. Rodríguez-Macías, F. Fernández, R. Domínguez-Castro, J. Huertas, and A. Rodríguez-Vázquez, "Global design of analog cells using statistical optimization techniques," *Analog integrated circuits and signal processing*, vol. 6, no. 3, pp. 179–195, 1994.

[15] B. Liu, F. Fernández, G. Gielen, R. Castro-López, and E. Roca, "A memetic approach to the automatic design of high-performance analog integrated circuits," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 14, no. 3, p. 42, 2009.

[16] 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.

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

[18] C. Rasmussen, "Gaussian processes in machine learning," *Advanced Lectures on Machine Learning*, pp. 63–71, 2004.

[19] 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.

[20] J. Sacks, W. Welch, T. Mitchell, and H. Wynn, "Design and analysis of computer experiments," *Statistical science*, vol. 4, no. 4, pp. 409–423, 1989.

[21] V. R. Joseph, Y. Hung, and A. Sudjianto, "Blind kriging: A new method for developing metamodels," *Journal of mechanical design*, vol. 130, p. 031102, 2008.

[22] I. Couckuyt, A. Forrester, D. Gorissen, F. De Turck, and T. Dhaene, "Blind kriging: Implementation and performance analysis," *Advances in Engineering Software*, vol. 49, pp. 1–13, 2012.

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

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

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

[26] W. L. Chan and J. R. Long, "A 58–65 GHz neutralized CMOS power amplifier with PAE above 10% at 1-V supply," *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 3, pp. 554–564, 2010.

[27] J. Chen and A. M. Niknejad, "A compact 1V 18.6 dBm 60GHz power amplifier in 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*. IEEE, 2011, pp. 432–433.

[28] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer methods in applied mechanics and engineering*, vol. 186, no. 2, pp. 311–338, 2000.

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

[30] B. Tessema and G. Yen, "A self adaptive penalty function based algorithm for constrained optimization," in *Evolutionary Computation, 2006. IEEE Congress on*. IEEE, 2006, pp. 246–253.

[31] 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.

[32] G. Yu and P. Li, "Yield-aware analog integrated circuit optimization using geostatistics motivated performance modeling," in *Computer-Aided Design, 2007. ICCAD 2007. IEEE/ACM International Conference on*. IEEE, 2007, pp. 464–469.

[33] ——, "Yield-aware hierarchical optimization of large analog integrated circuits," in *Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design*. IEEE Press, 2008, pp. 79–84.

[34] B. Liu, Y. He, P. Reynaert, and G. Gielen, "Global optimization of integrated transformers for high frequency microwave circuits using a Gaussian process based surrogate model," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011*. IEEE, 2011, pp. 1–6.

[35] D. M. Pozar, *Microwave engineering*. John Wiley & Sons, 2009.

[36] J.-W. Lai and A. Valdes-Garcia, "A 1V 17.9 dBm 60GHz power amplifier in standard 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*. IEEE, 2010, pp. 424–425.

[37] M. Boers, "A 60GHz transformer coupled amplifier in 65nm digital CMOS," in *Radio Frequency Integrated Circuits Symposium (RFIC), 2010 IEEE*. IEEE, 2010, pp. 343–346.

[38] B. Martineau, V. Knopik, A. Siligaris, F. Gianesello, and D. Belot, "A 53-to-68GHz 18dBm power amplifier with an 8-way combiner in standard 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*. IEEE, 2010, pp. 428–429.

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. He is currently a lecturer (Assistant Professor) with Department of Computing, Glyndwr University, U.K. From October 2012 to March 2013, he was a Humboldt research fellow and was working with Technical University of Dortmund, 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 30 papers in international journals and conference proceedings. Dr. Liu 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.

Dixian Zhao received the B.Sc. degree in microelectronics from Fudan University, Shanghai, China, in 2006, and the M.Sc. degree in microelectronics from the Delft University of Technology, Delft, The Netherlands, in 2009. Currently, he is pursuing the Ph.D. degree with Katholieke Universiteit Leuven, Leuven, Belgium. From 2005 to 2007, he was with the Auto-ID Laboratory, Fudan University, designing the nonvolatile memory for passive radio-frequency identification tags. From 2008 to 2009, he was an Intern with Philips Research, Eindhoven, The Netherlands, where he designed a 60 GHz beamforming transmitter for the presence of detection radar. From 2009 to 2010, he was with the Delft University of Technology, where he developed the 94 GHz wideband receiver for imaging radar. His current research interests include radio frequency and millimeter-wave integrated circuit design for wireless communications.

Patrick Reynaert was born in Wilrijk, Belgium, in 1976. He received the Master of Industrial Sciences in Electronics (ing.) from the Karel de Grote Hogeschool, Antwerpen, Belgium in 1998 and both the Master of Electrical Engineering (ir.) and the Ph.D. in Engineering Science (dr.) from the Katholieke Universiteit Leuven (K.U.Leuven), Belgium in 2001 and 2006 respectively. From 2001 to 2006, he was a Teaching and Research Assistant within the MICAS research group of the Department of Electrical Engineering (ESAT), K.U.Leuven, Belgium. During 2006-2007, he was a post-doctoral researcher at the Department of Electrical Engineering and Computer Sciences of the University of California at Berkeley. During the summer of 2007, he was a visiting researcher at Infineon, Villach, Austria. Since October 2007, he is an Associate Professor at the Katholieke Universiteit Leuven, department of Electrical Engineering (ESAT) and a staff member of the ESAT-MICAS research group. His research interests are in CMOS power amplifiers and in mm-wave CMOS integrated circuits.

Georges G. E. Gielen (S'87-M'92-SM'99-F'02) 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.