

Journal Article

**A Parallel Surrogate Model Assisted Evolutionary Algorithm for  
Electromagnetic Design Optimization**

Akinsolu, M. O., Liu, B., Grout, V., Lazaridis, P. I., Mognaschi, M. E. and Di Barba, P

This article is published by IEEE. The definitive version of this article is available at:  
<https://ieeexplore.ieee.org/abstract/document/8673708/authors#authors>

---

**Recommended citation:**

Akinsolu, M. O., Liu, B., Grout, V., Lazaridis, P. I., Mognaschi, M. E. and Di Barba, P (2019) 'A Parallel Surrogate Model Assisted Evolutionary Algorithm for Electromagnetic Design Optimization', IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 3, no. 2, pp. 93-105, 2019. doi: [10.1109/TETCI.2018.2864747](https://doi.org/10.1109/TETCI.2018.2864747)

# A Parallel Surrogate Model Assisted Evolutionary Algorithm for Electromagnetic Design Optimization

Mobayode O. Akinsolu *Member, IEEE*, Bo Liu *Senior Member, IEEE*, Vic Grout, Pavlos I. Lazaridis *Senior Member, IEEE*, Maria Evelina Mognaschi *Member, IEEE*, Paolo Di Barba *Member, IEEE*

**Abstract**—Optimization efficiency is a major challenge for electromagnetic (EM) device, circuit and machine design. Although both surrogate model-assisted evolutionary algorithms (SAEAs) and parallel computing are playing important roles in addressing this challenge, there is little research that investigates their integration to benefit from both techniques. In this paper, a new method, called parallel SAEA for electromagnetic design (PSAED), is proposed. A state-of-the-art SAEA framework, surrogate model-aware evolutionary search, is used as the foundation of PSAED. Considering the landscape characteristics of EM design problems, three differential evolution mutation operators are selected and organized in a particular way. A new SAEA framework is then proposed to make use of the selected mutation operators in a parallel computing environment. PSAED is tested by a micromirror and a dielectric resonator antenna as well as four mathematical benchmark problems of various complexity. Comparisons with state-of-the-art methods verify the advantages of PSAED in terms of efficiency and optimization capacity.

**Index Terms**—Electromagnetic design optimization; Electromagnetic design, Surrogate-model-assisted evolutionary algorithm; Computationally expensive optimization; Gaussian process; Differential evolution

## I. INTRODUCTION

Electromagnetic devices, circuits and machines are playing important roles in modern industry. Typical examples include antennas, filters, radio frequency circuits and systems, magnetic actuators and various kinds of electromechanical systems. With increasingly stringent design specifications and shortened time-to-market, global and local optimization methods are replacing the traditional trial-and-error design method [1], [2], [3]. Amongst these methods, evolutionary algorithms (EAs) such as differential evolution (DE) [4] and particle swarm optimization (PSO) [5] tend to play a leading role because of their global optimization capability, not requiring an initial design and robustness [1], [6], [7], [8], [9].

However, obtaining optimal designs in a reasonable optimization time becomes a challenge. To obtain an accurate performance estimation, numerical technique-based electromagnetic (EM) simulations are often necessary (e.g., employing

finite element analysis, method of moments), which are computationally expensive. Moreover, standard global optimization algorithms often need a large number of such simulations to get the optimum [10]. As a result, the optimization time can be very long or even prohibitive. To address this challenge, aside from some problem-specific techniques (e.g., analytical models), two major efficiency improvement methods include parallel computing and surrogate model-based optimization.

Employing parallel computing in an EA process is straightforward because the simulations of candidate designs are independent of each other in most cases. In recent years, the development of cloud computing enables EM designers to use distributed workers and memory for simulating multiple candidate designs simultaneously. Even in a standalone multicore desktop workstation, it is possible to parallelize several simulations, which is supported by some widely used simulation tools. However, due to the computing overhead of EM/multiphysics simulations and the financial cost of using shared computing resources, the number of candidate designs being parallelly simulated is often only a few in many cases [11]. Hence, although alleviated, the challenge in optimization efficiency still remains.

Another popular method is surrogate model-based optimization [12], [13], [14]. Although some novel surrogate model-based optimization methods aim at improving the optimization capacity [15], most of them aim at efficiency improvement. They employ surrogate models to replace computationally expensive exact function evaluations (e.g., EM simulations) so as to significantly reduce the computational cost. Surrogate models are computationally cheap approximation models of the exact evaluations, which are often constructed by statistical learning techniques. The search engine can be infill sampling, local search and global search. When EA becomes the search engine, surrogate model-assisted evolutionary algorithms (SAEAs) are constructed. In SAEA research, there is a critical trade-off between the surrogate model quality and the efficiency (reflected by the necessary number of exact evaluations), and the method to find an appropriate trade-off is called surrogate model management [16]. Based on different surrogate model management routines, there are various kinds of SAEAs.

The surrogate model-aware evolutionary search (SMAS) framework is a state-of-the-art SAEA framework [17], [18]. In particular, it has been widely applied in EM design optimization. Several state-of-the-art mm-wave integrated circuit, antenna, filter and microelectromechanical system design optimization methods borrow some idea of the SMAS framework

M. O. Akinsolu and V. Grout are with Department of Computing, Wrexham Glyndwr University, U.K. (e-mail: m.o.akersolu@ieee.org, v.grout@glyndwr.ac.uk).

B. Liu (Corresponding Author) is with Department of Computing, Wrexham Glyndwr University, U.K. and School of Electrical, Electronic and System Engineering, The University of Birmingham (e-mail: b.liu.3@bham.ac.uk, b.liu@glyndwr.ac.uk).

P. Lazaridis is School of Computing and Engineering, University of Huddersfield, U.K. (e-mail: p.lazaridis@hud.ac.uk).

M. E. Mognaschi and P. Di Barba are with School of Electrical, Computer and Biomedical Engineering, University of Pavia, Italy. (e-mails: eve.mognaschi@unipv.it, paolo.dibarba@unipv.it).

[14], [19], [20], [21]. For these EM design problems, the SMAS framework-assisted SAEAs outperform several other state-of-the-art SAEAs in terms of efficiency and optimization quality [14], [19], [20]. In addition, algorithms based on SMAS are becoming industry standard methods for antenna design. However, the SMAS framework also requires a long optimization time when the fidelity of the EM model is very high or several fields need to be analyzed in a single simulation (i.e., each simulation is computationally very expensive) [22].

It is a natural idea to combine the SMAS framework and parallel computing to address the challenge of optimization efficiency. However, this is NOT trivial. The SMAS framework is sequential (i.e., in contradiction to parallelization), and a single simulation is carried out in each iteration. The next candidate design to be simulated is then decided by the updated surrogate model [17], [18].

The goal of this paper is to enable the SMAS framework to work in a parallel computing environment for efficient EM design optimization. The following problems will be investigated: (1) What are the landscape characteristics of EM design problems and what search operators are needed for effective optimization under the SMAS framework; (2) How to obtain an optimal trade-off between the surrogate model quality and the efficiency for the targeted problem when considering parallel simulations; and (3) How to develop a parallel SAEA framework taking full advantage of SMAS? Based on the above investigation, a new algorithm, called parallel surrogate model-assisted evolutionary algorithm for electromagnetic design (PSAED), is proposed. In particular, PSAED aims to:

- Obtain at least comparable results with methods which directly embed EM simulations in a standard EA (often considered as the benchmark in terms of solution quality);
- Support parallel computing environments, where a few candidate designs can be simulated simultaneously;
- Significantly enhance the efficiency of standard parallel simulation-based EAs;
- Be generic in handling EM design optimization problems without the need for initial solutions or an ad-hoc process.

The remainder of the paper is organized as follows: Section II presents the basic techniques. Section III elaborates the PSAED method, including landscape characteristics analysis of EM design problems, the behavioral study of SMAS for the targeted design landscapes, mutation strategy selection and the new parallel SAEA framework. Section IV presents the performance of PSAED using a micromirror, a dielectric resonator antenna and four mathematical benchmark problems. Comparisons with parallel DE, parallel PSO and parallel SMAS are carried out. The concluding remarks are provided in Section V.

## II. BASIC TECHNIQUES

### A. Gaussian Process Surrogate Modeling

Gaussian Process (GP) is the surrogate modeling method chosen for PSAED [23], which is widely used in efficient EM design exploration [14], [13], [24]. GP works as follows [23]:

For a given set of  $n$  observations  $x = (x^1, \dots, x^n)$  and  $y = (y^1, \dots, y^n)$ , GP modeling assumes that the objective function  $y(x)$  is a sample of a Gaussian distributed stochastic process with mean  $\mu$  and variance  $\sigma$ . Based on available observations, GP predicts a new function value  $y(x)$  at some new design point  $x$ . If  $y(x)$  is a continuous function and  $x_i$  and  $x_j$  are highly correlated,  $y(x_i)$  and  $y(x_j)$  should be close. In this work, the Gaussian correlation function stated in (1) describes the correlation between two variables.

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

where  $d$  is the dimension of  $x$  and  $\theta_l$  is the correlation parameter which determines the extent of correlation reduction when  $x_i^l$  moves in the  $l$  direction. The smoothness of the function is related to  $p_l$  with respect to  $x^l$ . The hyper-parameters  $\theta_l$  and  $p_l$  are determined by maximizing the likelihood function in (2).

$$\frac{1}{(2\pi\sigma^2)^{n/2} \sqrt{\det(R)}} \exp \left[ -\frac{(y - \mu I)^T R^{-1} (y - \mu I)}{2\sigma^2} \right] \quad (2)$$

where  $R$  is a  $n \times n$  matrix and  $I$  is a  $n \times 1$  vector having all its elements as unity. By maximizing the likelihood function that  $y = y^i$  at  $x = x^i$  ( $i = 1, \dots, n$ ) and handling the prediction uncertainty based on the best linear unbiased prediction, the function value  $y(x^*)$  at a new point  $x^*$  can be predicted as:

$$\hat{y}(x^*) = \mu + r^T R^{-1} (y - \mu I) \quad (3)$$

where

$$\begin{aligned} R_{i,j} &= \text{Corr}(x_i, x_j), i, j = 1, 2, \dots, n \\ r &= [\text{Corr}(x^*, x_1), \text{Corr}(x^*, x_2), \dots, \text{Corr}(x^*, x_n)] \\ \hat{\mu} &= (I^T R^{-1} I)^{-1} I^T R^{-1} y \end{aligned} \quad (4)$$

For the surrogate model, the mean square error value of the prediction uncertainty is stated in (5).

$$\hat{\sigma}^2(x^*) = \hat{\sigma}^2 [I - r^T R^{-1} r + (I - r^T R^{-1} r)^2 (I^T R^{-1} I)^{-1}] \quad (5)$$

where

$$\hat{\sigma}^2 = (y - I\hat{\mu})^T R^{-1} (y - I\hat{\mu}) n^{-1} \quad (6)$$

The computational complexity of GP modeling is  $O(N_{it} K^3 d)$  [25], where  $N_{it}$  is the number of iterations spent in hyper-parameter optimization and  $K$  is the number of training data points. The most critical factor is  $K$ , which is affected by  $d$  in order to construct a reliable surrogate model. Considering the dimensionality of the targeted problem (Section III (A)), the GP modeling time is often short.

To evaluate the quality of a candidate design in terms of the predicted value in (3) and the prediction uncertainty in (5), several prescreening methods can be used. The lower confidence bound (LCB) method [25] is adopted in PSAED. If the objective function  $y(x)$  has a predictive distribution of  $N(\hat{y}(x), \hat{\sigma}^2(x))$ , an LCB prescreening of  $y(x)$  can be defined as follows:

$$\begin{aligned} y_{lcb}(x) &= \hat{y}(x) - \omega \hat{\sigma}(x) \\ \omega &\in [0, 3] \end{aligned} \quad (7)$$

where  $\omega$  is a constant, which is often set to 2 to ensure a balance between exploration and exploitation [25]. In this work, the GP modeling is implemented using the ooDACE toolbox [26].

### B. The DE Algorithm

The DE algorithm [4] is adopted as the search engine in PSAED, which works as follows:

Let  $P$  be a population composed of a number of individual solutions  $x = (x_1, \dots, x_d) \in R^d$ . To create a child solution  $u = (u_1, \dots, u_d)$  for  $x$ , firstly, mutation occurs to produce a donor vector:

$$v^i = x^{r_1} + F \cdot (x^{r_2} - x^{r_3}) \quad (8)$$

where  $x^{r_1}$ ,  $x^{r_2}$  and  $x^{r_3}$  are three mutually exclusive solutions randomly selected from  $P$ ;  $v^i$  is the  $i^{th}$  mutant vector;  $F \in (0, 2]$  is a control parameter, often called the scaling factor. The mutation strategy in (8) is called DE/rand/1.

The crossover operator is then applied to produce the child solution  $u$ :

- 1 Randomly select a variable index  $j_{rand} \in \{1, \dots, d\}$ ,
- 2 For each  $j = 1$  to  $d$ , generate a uniformly distributed random number  $rand$  from  $(0, 1)$  and set:

$$u_j = \begin{cases} v_j, & \text{if } (rand \leq CR) | j = j_{rand} \\ x_j, & \text{otherwise} \end{cases} \quad (9)$$

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

In this study, four other popular DE mutation strategies stated in (10) to (13) are involved:

- (1) mutation strategy: DE/best/1

$$v^i = x^{best} + F \cdot (x^{r_1} - x^{r_2}) \quad (10)$$

where  $x^{best}$  is the best candidate solution in the current population  $P$ .

- (2) mutation strategy: DE/current-to-best/1

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

where  $x^i$  is the  $i^{th}$  vector in the current population.

- (3) mutation strategy: DE/best/2

$$v^i = x^{best} + F \cdot (x^{r_2} - x^{r_3}) + F \cdot (x^{r_4} - x^{r_5}) \quad (12)$$

where  $x^{r_4}$  and  $x^{r_5}$  are two different solutions randomly selected from  $P$  and are different from  $x^{r_1}$ ,  $x^{r_2}$  and  $x^{r_3}$ .

- (4) mutation strategy: DE/rand/2

$$v^i = x^{r_1} + F \cdot (x^{r_2} - x^{r_3}) + F \cdot (x^{r_4} - x^{r_5}) \quad (13)$$

(8), (10), (11), (12) and (13) are arguably the most widely used DE mutation strategies.

### C. The SMAS Framework

There are several versions of the SMAS framework (e.g., [17], [19], [18]). The following version is used because it shows the overall best performance for various kinds of EM devices, circuits and machines.

- Step 1:** Sample  $\alpha$  (often a small number) solutions from the design space  $[a, b]^d$  ( $a$  and  $b$  are the lower and upper bounds of the design variables, respectively;  $d$  is the number of design variables) using the Latin Hypercube sampling method [27]. Evaluate the real function values of all these solutions and let them form the initial database.
- Step 2:** If a preset stopping criterion is met, output the best solution from the database; otherwise go to Step 3.
- Step 3:** Select the  $\lambda$  best solutions from the database to form a population  $P$ .
- Step 4:** Apply the DE/current-to-best/1 operator (11) to  $P$  to create  $\lambda$  new child solutions.
- Step 5:** For each child solution, select  $\tau$  nearest samples (based on Euclidean distance) as the training data points and construct a local GP surrogate model. Prescreen the child solutions generated in Step 4 using the LCB method in Section II (A).
- Step 6:** Evaluate the real function value of the estimated best child solution from Step 5. Add this evaluated solution and its objective function value to the database. Go back to Step 2.

There are defined parameter setting rules for SMAS. When following the rules, SMAS shows high robustness for various kinds of problems [18], [20]. Compared to SAEAs using a standard EA structure, the key advantage of SMAS is the improved locations of training data points. With the same number of training data points, it is intuitive that using training data points located near to the points waiting to be predicted (child population) can obtain better surrogate models and prediction results.

In SMAS, the  $\lambda$  current best candidate solutions form the parent population (it is reasonable to assume that the search focuses on the promising region) and the best candidate based on prescreening in the child population is selected to replace the worst one in the parent population. Hence, only at most one candidate solution is changed in the parent population in each iteration. Thus, the best candidate in the child solutions in several consecutive iterations may be quite near to each other (they will then be simulated and are used as training data points). Therefore, the training data points describing the current promising region can be much denser compared to those generated by a standard EA population updating mechanism, which may spread in different regions of the design space. The latter may lead to insufficient training data points around the candidate solutions to be prescreened [18], [28].

## III. THE PSAED METHOD

### A. Landscape Characteristics of EM Design Problems

Because real-world EM design problems are computationally expensive, they are not fit for a behavioral study that

requires a number of runs. Thus, mathematical benchmark problems which have similar landscape characteristics are selected. Size and complexity are two main aspects to be considered in selection. The number of design variables is often below 10 for electromagnetic machines [29], [21], below or around 10 for microwave antennas, filters and other devices [30], [24] and 10 to 30 for circuits [19], [31]. Scalability is an advantage of the SMAS framework and its capacity to handle 30-dimensional complex problems is shown in [20]. To cover most EM design problems, 10-dimensional mathematical benchmark problems are used.

Selecting mathematical benchmark problems with appropriate complexity is essential. Much EA research assumes a sufficient number of function evaluations, and very complex landscapes are used to challenge the search algorithms. However, in surrogate-based optimization research, different surrogate model management methods have different trade-offs between the complexity of the landscape that it can handle and the efficiency. It is clear that, to handle complex landscapes, more training data points are needed so as to obtain reliable surrogate models, which decreases the efficiency.

The landscapes of electromagnetic machines (e.g., EM actuators) are often not complex and (multi-start) local optimization works well in many cases [32], [33]. Antenna and microwave device and circuit landscapes are shown to be multimodal [1], [34]. However, very rugged landscapes (e.g., the Rastrigin function [35]), which tend to be discontinuous rarely appear. This is because EM simulations are, in fact, solving Maxwell's equations and such a landscape is not typically generated by partial differential equations. Based on this, we use the Ellipsoid function (14) [36] to represent simple landscapes and the Ackley function (15) [36] to represent multimodal landscapes. The reason to select the Ackley function is that the more complex the landscape, the more speed improvement SMAS (or most SAEAs) has compared to standard EAs [14]. To the best of our knowledge, the speed improvement of SMAS is often lower when using real-world problems than using the Ackley function.

$$F(x) = \sum_{i=1}^d ix_i^2 \quad (14)$$

$$x \in [-20, 20], i = 1, \dots, 10$$

$$F(x) = -20e^{-0.2\sqrt{\frac{1}{d}\sum_{i=1}^d x_i^2}} - e^{\frac{1}{d}\sum_{i=1}^d \cos(2\pi x_i)} + 20 + e \quad (15)$$

$$x \in [-32, 32], i = 1, \dots, 10$$

### B. Behavioral Analysis of DE Mutation Strategies under the SMAS Framework

Although behaviors of the five standard DE mutation strategies are generally known, they have not been well studied under the SMAS framework, which is essential for designing PSAED. 30 independent runs are carried out with the setting of  $\alpha = 5 \times d$ ,  $\lambda = 50$ ,  $\tau = 8 \times d$ ,  $F = 0.8$  and  $CR = 0.8$ , which is in line with the SMAS parameter setting rules [18], [14]. The computing budget is 1,000 function evaluations in each run. The Hopkins test [37] result (average of H-measure) is shown in Fig. 1. The H-measure shows the clustering tendency

of the optimization process: 1 indicates full convergence and 0.5 indicates regularly spaced points. The setting of H-measure computation follows [38].

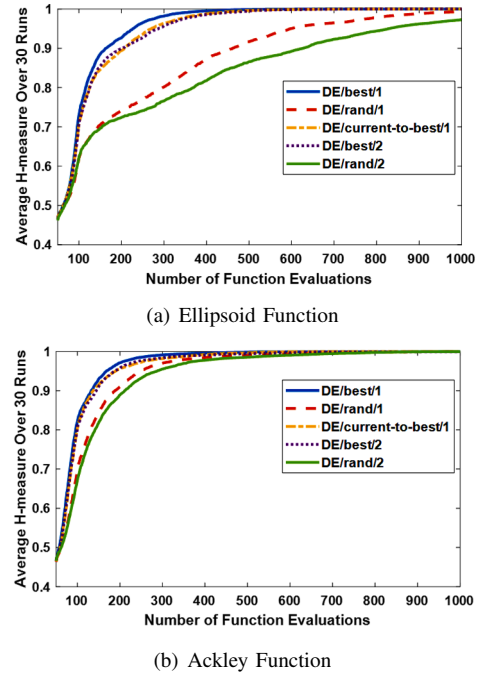


Figure 1. H-measure of the five DE mutation strategies for the Ellipsoid and Ackley functions (SMAS framework)

It can be seen that under the SMAS framework: (1) The rank of convergence speed (from high to low) is DE/best/1, DE/current-to-best/1, DE/best/2, DE/rand/1, DE/rand/2. (2) The convergence speed varies a lot and forms two distinct clusters. The first three strategies are several times faster than the last two strategies. In particular, 1000 evaluations are not enough for the second cluster to converge for the Ellipsoid function. (3) In the first cluster (i.e., fast convergence cluster), the convergence speed is similar. In the second cluster (i.e., slow convergence cluster), a non-negligible difference is shown.

In terms of solution quality, the box plots are shown in Fig. 2. It can be seen that: (1) For the Ellipsoid function, all mutation strategies (tend to) converge to the global optimal point. Note that the outliers of the DE/rand/1 and DE/rand/2 are not the result of being trapped at local optima (the Ellipsoid function is unimodal), but because of the insufficient number of function evaluations. (2) For the Ackley function, under the SMAS framework, some of the runs of DE/best/1, DE/current-to-best/1, DE/best/2 are trapped at local optima, although the average and median values are reasonably good (Table I). Note that such a result is often considered as a success in computational intelligence research, but greater robustness is highly desirable in real-world engineering optimization. In contrast, DE/rand/1 and DE/rand/2 are seldom trapped at local optima especially when using DE/rand/2.

Therefore, behavioral study shows that, under the SMAS framework, there is no winning DE mutation strategy which has good enough exploration ability, while at the same time being among the efficient strategies for the targeted EM

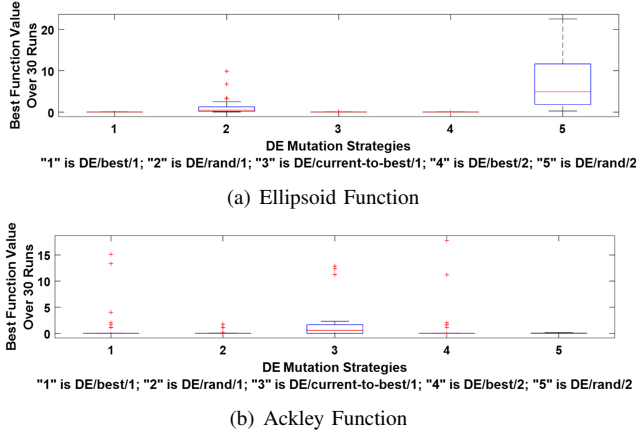


Figure 2. Box plots of five DE mutation strategies for the Ellipsoid and Ackley functions (SMAS framework)

Table I  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES USING DIFFERENT DE MUTATION STRATEGIES (ACKLEY FUNCTION) (OVER 30 RUNS)

Strategy	Best	Worst	Mean	Median	Std.
DE/best/1	1.15e-04	15.1476	1.2841	3.85e-04	3.6391
DE/rand/1	9.71e-04	1.7569	0.1900	0.0082	0.4597
DE/c-to-b/1	2.44e-04	12.8622	1.8521	0.5915	3.5976
DE/best/2	1.53e-06	17.8114	1.1835	8.87e-04	3.7676
DE/rand/2	0.0064	0.1277	0.0487	0.0348	0.0344

design problems. Therefore, two natural questions are: (1) Is it possible to employ multiple mutation strategies to combine the advantages and remedy the shortcomings? (2) Can employing multiple mutation strategies be in line with the parallel computing environment? These questions will be answered in the next subsection.

### C. The PSAED Method

1) *Challenges for introducing parallel simulations in SMAS*: As mentioned above, the SMAS framework is sequential. A quick attempt at implementing the SMAS framework in a parallel computing environment is to simulate the top  $k$  (instead of the top 1) candidate designs in Step 6 (Section II (C)) parallelly in each iteration. In a similar fashion to simulate  $k$  candidates in parallel in standard EAs, a reduction of  $k$  times computing overhead should be observed if this approach is effective. However, using this method for the 10-dimensional Ackley function shows a 2.1 times speed improvement for  $k = 4$  (Fig. 3).

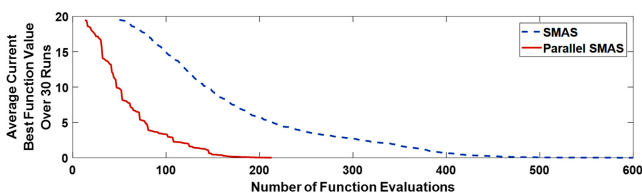


Figure 3. Comparison of convergence trends (DE/current-to-best/1, Ackley function, average of 30 runs)

By observing the generated candidate solutions in the optimization process, it can be found that: (1) The predicted best candidates in most iterations are often the truly best ones with very few other candidates having similar optimality. (2) The best candidate may remain unchanged over a consecutive number of iterations (usually, 10 - 30). Therefore, some of the  $k$  candidates that are simulated in each iteration are suboptimal solutions when  $k = 4$ . Simulating suboptimal solutions can neither directly improve the result nor improve the present surrogate model in the optimal region. This implies that generating more diverse and high-quality candidate solutions can help the parallelization of the SMAS framework. Using multiple DE mutation strategies can possibly support this objective.

2) *Using multiple mutation strategies*: Under the SMAS framework, for the Ackley function, the average and median values when using the DE/best/1 mutation strategy are reasonably acceptable (Table I). However, a few outliers show poor results. Hence, an initial idea is to moderately improve the exploration ability of DE/best/1 to reduce the outliers while keeping the convergence speed. According to Fig. 2(b), the DE/current-to-best/1 strategy shows the least outliers among the fast convergence cluster. Hence, the combination of DE/best/1 and DE/current-to-best/1 is investigated first. The following method is used.

**Step 1:** Implement Steps 1 to 3 of the SMAS framework (Section II (C)).

**Step 2:** Set the rates of using DE/best/1 and DE/current-to-best/1,  $\hat{R}_1$  and  $\hat{R}_2$ , respectively.

**Step 3:** Use roulette wheel selection [39] based on the given rates to determine a DE mutation strategy. Generate  $\lambda$  child solutions using the selected strategy to construct a child population. Repeat the above process to construct the other child population.

**Step 4:** Implement Step 5 of the SMAS framework.

**Step 5:** Evaluate the real function value of the estimated best child solution from each population. Add the evaluated solutions and their objective function values to the database. Start the next iteration.

Three kinds of combinations are (1)  $\hat{R}_1 = 0.5$  and  $\hat{R}_2 = 0.5$ , (2)  $\hat{R}_1 = 1$  and  $\hat{R}_2 = 0$ , and (3)  $\hat{R}_1 = 0$  and  $\hat{R}_2 = 1$ . The number of independent runs and parameter settings are the same as Section III (B). The computing budget is 500 and 300 parallel function evaluations, respectively, to make both problems converge. For convergence, we mean that the current best candidate solution shows very minor improvement in 50 consecutive iterations.

The efficiency of the three mutation strategies in the fast convergence cluster is very similar (Fig. 1). Hence, the convergence speed of the three kinds of combinations is almost the same, which is verified by H-measure analysis. In terms of solution quality, the box plots are shown in Fig. 4. Table II and Table III show the statistics.

The following observations can be made: (1) Combinations of strategies show significant improvements compared to using a single strategy in Section III (B), even for a combination of the same strategy. (2) The combination of 50% DE/best/1 and 50% DE/current-to-best/1 shows the largest improvement

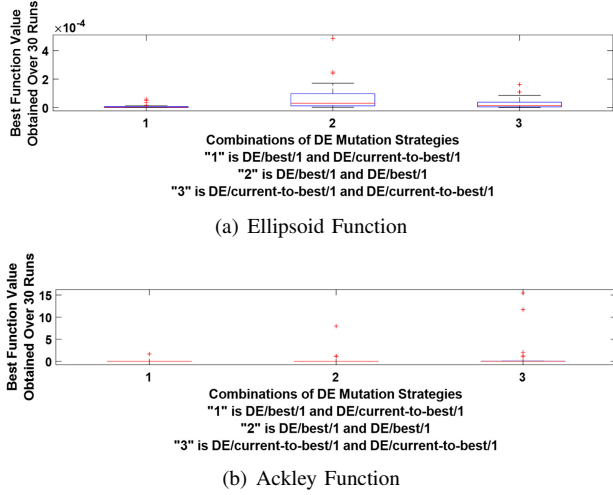


Figure 4. Box plots of the combinations of DE/best/1 and DE/current-to-best/1 for the Ellipsoid and Ackley functions

Table II  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES USING COMBINATIONS OF DE/BEST/1 AND DE/CURRENT-TO-BEST/1 AT DIFFERENT RATES (ELLIPSOID FUNCTION) (OVER 30 RUNS)

Combination	Best	Worst	Mean	Median	Std.
DE/best/1 (50%)	3.72e-07	5.96e-05	8.39e-06	2.02e-06	1.45e-05
DE/c-to-b/1 (50%)					
DE/best/1 (100%)	2e-06	4.85e-04	7.6e-05	3.04e-05	1.03e-04
DE/c-to-b/1 (100%)					
DE/c-to-b/1 (100%)	1.21e-06	1.62e-04	2.82e-05	1.31e-05	3.85e-05

compared with alternatives. It is clear that the outliers are significantly reduced. These two observations are understandable because the two mutation strategies produce diverse optimal solutions compared to only using a single strategy. They compensate each other for improvements in both the objective function value and the surrogate model quality. When combining the same strategy, more diverse candidate solutions are also generated because  $x^{r1}$  and  $x^{r2}$  are randomly selected, although the strength is not as good as the combination of DE/best/1 and DE/current-to-best/1.

Based on this observation, a natural question is to investigate the performance when introducing a mutation strategy with

Table III  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES USING COMBINATIONS OF DE/BEST/1 AND DE/CURRENT-TO-BEST/1 AT DIFFERENT RATES (ACKLEY FUNCTION) (OVER 30 RUNS)

Combination	Best	Worst	Mean	Median	Std.
DE/best/1 (50%)	2.1e-04	1.6462	0.0567	7.6287e-04	0.3002
DE/c-to-b/1 (50%)					
DE/best/1 (100%)	1.67e-04	7.9485	0.3813	7.11e-04	1.4718
DE/c-to-b/1 (100%)					
DE/c-to-b/1 (100%)	0.0034	15.5186	1.4293	0.0135	3.5270

higher exploration ability (the slow convergence cluster in Section III (B)). Will the overall exploration ability be further improved? Will the efficiency be largely decreased?

Although DE/rand/1 and DE/rand/2 are several times slower than the mutation strategies in the fast convergence cluster under the standard SMAS framework, it may not necessarily slow down the convergence when introducing them into the combination. According to the trade-off between the exploration ability and the probability of finding the correct search direction [40], many of the candidate solutions generated by slow convergence strategies are not in the correct directions in standard SMAS. However, when using the foundation constructed by DE/best/1 and DE/current-to-best/1, most of the candidate solutions are near the optimal region. Hence, many candidate solutions generated by the slow convergence cluster can be considered as exploring the optimal region instead of the whole space. When the landscape is simple, the exploration of the optimal region largely improves the surrogate model, which, in contrast, has the possibility to improve the efficiency. For complex landscapes, the exploration ability of the slow convergence cluster can also contribute to the overall exploration ability without considerably slowing down the convergence.

To verify the above assumption and decide the appropriate strategy between DE/rand/1 and DE/rand/2, the following experiments are carried out. It is not easy to predefine the rate of using different mutation strategies, which is problem dependent: the following self-adaptive framework is, therefore, proposed.

For EACH child population  $i = 1, 2, \dots, n$

**Step 1:** If the algorithm is within the learning period (the current number of iterations is smaller than a threshold  $L$ ), the rates of using DE/best/1 (10), DE/current-to-best/1 (11) and DE/rand/1 (8) or DE/rand/2 (13) are equal, that is  $\hat{R}_1 = \hat{R}_2 = \hat{R}_3 = \frac{100}{3}\%$ . Otherwise; use the rates in Step 5.

**Step 2:** Use roulette wheel selection [39] based on the rates to determine a DE mutation strategy and generate a child population ( $C_i$ ) having  $\lambda$  child solutions.

**Step 3:** Use the local GP surrogate models in Step 5 of the SMAS framework (Section II (C)) to predict all the candidate designs in  $C_i$ .

**Step 4:** Compare the predicted value of each solution in  $C_i$  and the current best solution (simulated value) of all populations. Add the number of solutions that are better than the current best solution to  $N_s$  (the number of successes of DE/best/1 (10), DE/current-to-best/1 (11) and DE/rand/1 (8) or DE/rand/2 (13)) and add  $\lambda$  to  $N_u$  (the number of uses of these mutation strategies).

End the loop

**Step 5:** Update the rates of using DE/best/1 (10), DE/current-to-best/1 (11) and DE/rand/1 (8) or DE/rand/2 (13) by  $N_s/N_u$ . Update the number of iterations.

Step 4 is essential in this self-adaptive method. If the predicted objective function value of a candidate solution

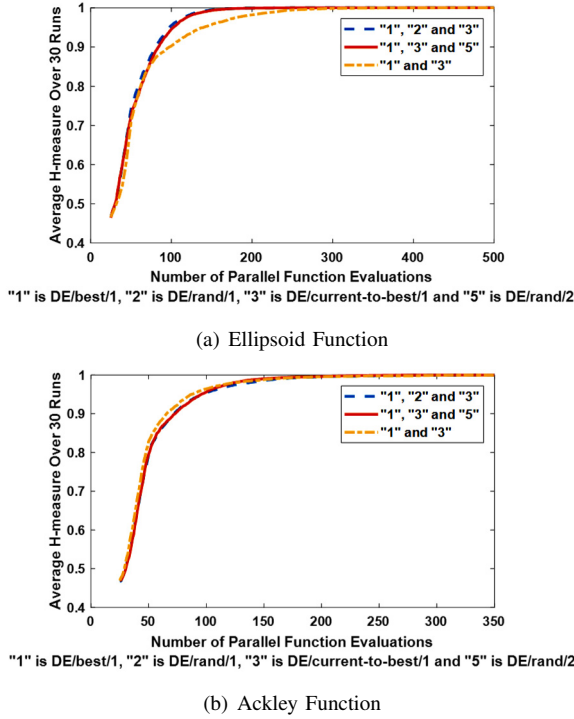


Figure 5. H-measure of the combinations of DE/best/1, DE/current-to-best/1 and DE/rand/1 or DE/rand/2 for the Ellipsoid and Ackley functions

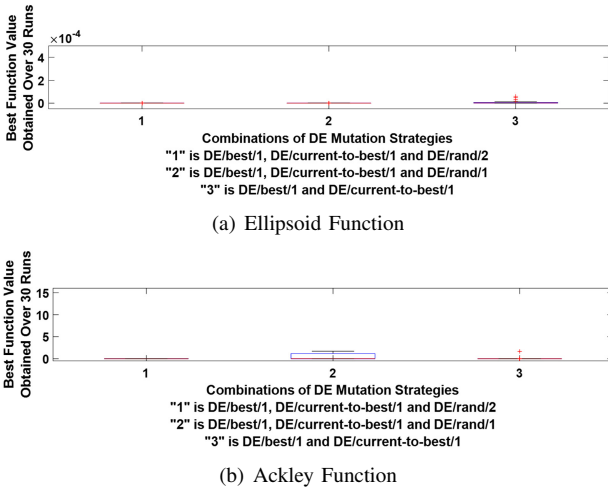


Figure 6. Box plots of the combinations of DE/best/1, DE/current-to-best/1 and DE/rand/1 or DE/rand/2 for the Ellipsoid and Ackley functions

generated by a certain DE mutation strategy is better than the best-so-far simulated objective function value, we assume that it is a successful case of the corresponding DE mutation strategy. The success rates then decide the probability of using different DE mutation strategies in future iterations.

30 independent runs are used and the result is shown in Fig. 5 and Fig. 6. It can be seen that the assumption is verified and DE/rand/2 is more appropriate than DE/rand/1. In particular, the efficiency of optimizing the Ellipsoid function is, in contrast, improved when including the slow convergence cluster. All the outliers are removed when using DE/best/1, DE/current-to-best/1 and DE/rand/2 mutation strategies.

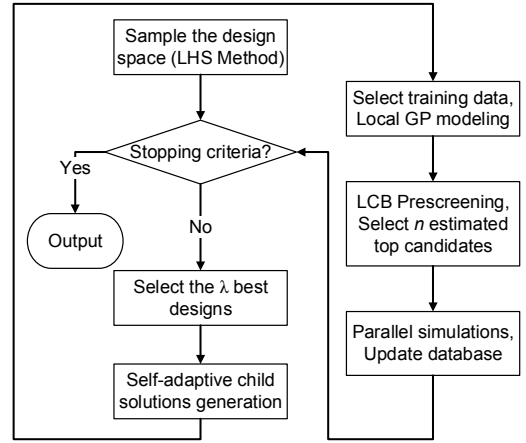


Figure 7. The flow diagram of PSAED

3) *The PSAED framework*: Based on the above, the flow diagram of PSAED is shown in Fig. 7. Assuming  $n$  candidate designs can be simulated in parallel, PSAED works as follows.

- Step 1:** Use the Latin Hypercube sampling method [27] to sample  $\alpha$  (often small) designs from the design space and simulate them to constitute the initial database.
- Step 2:** If a preset stopping criterion such as the maximum number of simulations is met, output the best design from the database; otherwise go to Step 3.
- Step 3:** Select the  $\lambda$  best designs from the database to form a population  $P$ .
- Step 4:** Apply the self-adaptive DE/best/1, DE/current-to-best/1 and DE/rand/2-based search (Section III (C)(2)) on  $P$  to create  $n$  child populations (each population has  $\lambda$  child solutions).
- Step 5:** For each candidate design in each population, construct a local GP surrogate model using the  $\tau$  nearest designs (based on Euclidean distance) from the database and their performance values as the training data points. (There are  $n \times \lambda$  GP models in total.)
- Step 6:** Prescreen the  $n \times \lambda$  child solutions generated in Step 4 using the GP models in Step 5 and the LCB method (7). Select the top  $n$  child solutions based on the LCB values.
- Step 7:** Simulate the estimated top  $n$  child solutions from Step 6 in parallel. Add them and their performance (via simulation) to the database. Go back to Step 2.

In terms of parameter settings, PSAED introduces only one new parameter  $L$  (learning period) compared to SMAS. Clearly, this parameter is not sensitive. For simplicity, it is set to 50 for all the problems. The parameter setting rules of SMAS [17], [18], [19] are still useful. In all the test cases, the following parameters are used:  $\alpha = 5 \times d$ ,  $\lambda = 50$ ,  $\tau = 8 \times d$ ,  $F = 0.8$ ,  $CR = 0.8$ .



#### IV. EXPERIMENTAL RESULTS AND COMPARISONS

In this section, two real-world examples, including a 7-variable magnetically actuated micromirror and a 7-variable hybrid dielectric resonator antenna (DRA), as well as four mathematical benchmark problems, which are 6- $d$  Hartmann (22), 10- $d$  Ackley (15), Ellipsoid (14) and Michalewics functions (23) [36], are used to demonstrate the PSAED method. The reference methods are parallel SMAS, parallel DE [4] and parallel PSO [5]. For all problems, we assume 3 candidate solutions can be simulated in parallel. Parallelizing DE and PSO is straightforward. For SMAS, the top 3 candidate solutions in each iteration are simulated in parallel.

The parameter setting of PSAED is described in the last section. The parameters for parallel SMAS and DE are the same as those used in PSAED for a fair comparison. Regarding parallel PSO, the swarm size is set to 50, which is the same as other reference methods. For other PSO parameters, MATLAB's default setting is used (cognitive and social parameters of 1.49 each, adaptive inertia weight with a boundary of [0.1, 1.1] and a minimum adaptive neighborhood size of 0.25). All the tests are run on a workstation with Intel 4-core i7 CPU and 24GB RAM and the time consumption is wall clock time.

##### A. Example 1

The first example is a magnetically actuated micromirror [2] with a layout shown in Fig. 8. The micromirror is analyzed by the finite element method. The finite element model of the micromirror has a depth ( $w$ ) of  $10\mu m$ , residual induction of the magnet of  $1.04T$ , and relative magnetic permeability of  $10^3$  and 1.1 for the plate and magnet, respectively. The magnetic vector potential ( $\vec{A}$ ) and magnetic induction ( $\vec{B}$ ) field are evaluated using (16) and (17), respectively.

$$\nabla^2 \vec{A} = -\mu_m \vec{J} - \nabla \times \vec{B}_o \quad (16)$$

where  $\vec{J}$  is the coil current,  $\mu$  is the magnetic permeability and  $\vec{B}_o$  is the magnetic remanence of the permanent magnet.

$$\vec{B} = \nabla \times \vec{A} \quad (17)$$

From (16) and (17), a typical finite-element mesh is composed of 100,000 triangles; second-order Lagrangian elements are considered. The torque is then computed based on the virtual work principle as follows:

$$T(X, \varphi) = \left. \frac{\partial W'}{\partial \varphi} \right|_{\varphi=30^\circ} \quad (18)$$

where  $W'$  is the co-energy calculated as:

$$W' = \frac{1}{2} \int_{\Omega} \frac{B^2}{\mu} d\Omega \quad (19)$$

The micromirror is modeled in MagNet and the time of each simulation varies (typically 2-10 minutes). Given the vector  $X$  of the design variables, the design exploration goal is to maximize the actuation torque ( $T$ ) for a mirror position ( $\varphi$ ) of  $30^\circ$ :

$$\underset{X}{\text{maximize}} T(X, \varphi) \text{ with } (\varphi = 30^\circ) \quad (20)$$

Table IV  
RANGES OF THE DESIGN VARIABLES (ALL SIZES IN  $\mu m$ ) FOR MICROMIRROR DESIGN EXPLORATION

Variables	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$
Lower bound	50	1000	25	25	100	300
Upper bound	150	2000	75	75	300	900

Table V  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES USING DIFFERENT METHODS (EXAMPLE 1, IN  $nNm$ )

Method	Best	Worst	Mean	Median	Std.
PSAED	1.015	1.015	1.015	1.015	1.9e-6
Parallel SMAS	0.966	0.866	0.922	0.928	0.031
Parallel DE	0.847	0.766	0.812	0.822	0.0414
Parallel PSO	0.977	0.877	0.913	0.883	0.0564

The parameters that can be adjusted include the magnet height ( $X_1$ ), the magnet length ( $X_2$ ), the magnet air-gap ( $X_3$ ), the conductor height ( $X_4$ ), the conductor length ( $X_5$ ) and the conductor air-gap ( $X_6$ ) at a fixed pulsed current density of  $5Am^{-2}$ . Their ranges are shown in Table IV. To make all methods converge, the computing budget is as follows: 250 parallel simulations for PSAED and SMAS, respectively, and 1250 parallel simulations for DE and PSO, respectively. 10 independent runs are carried out for PSAED and SMAS, respectively, while 3 independent runs are carried out for parallel DE and PSO, respectively, because more runs are not affordable.

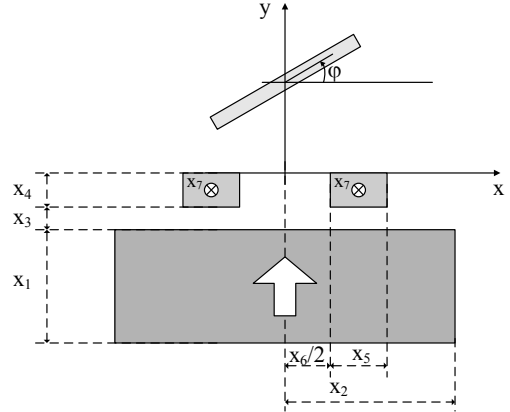


Figure 8. Layout of the magnetically actuated micromirror.

Optimization results are shown in Table V. A typical response for PSAED is shown in Fig. 9 with the obtained optimal design of  $X_1 = 15\mu m$ ,  $X_2 = 1086\mu m$ ,  $X_3 = 28.3\mu m$ ,  $X_4 = 30.5\mu m$ ,  $X_5 = 162\mu m$ ,  $X_6 = 484\mu m$ . The optimization time is about 25 hours. It can be seen that PSAED obtains very satisfactory results. Moreover, in all 10 runs, it finds almost the same optimal design, which is better than all designs that the reference methods find. Hence, a clear advantage over parallel SMAS, parallel DE and parallel PSO is shown in terms of design solution quality.

In terms of efficiency, PSAED uses around 168 parallel

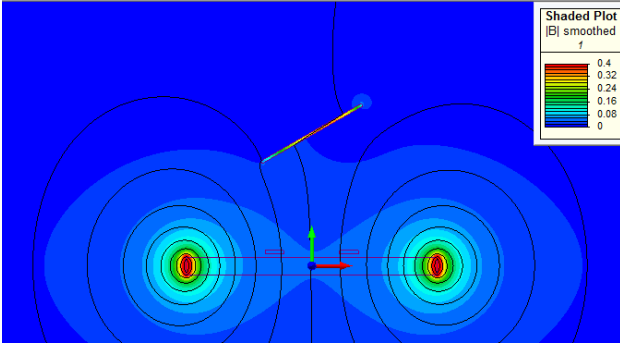


Figure 9. A typical magnetic field map of the optimized micromirror by PSAED

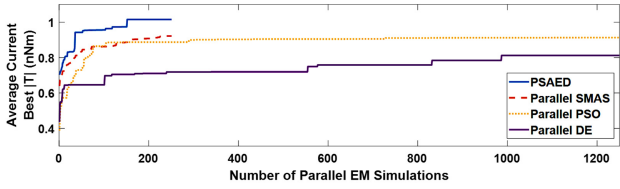


Figure 10. Convergence trends of PSAED (10 runs), parallel SMAS (10 runs), parallel DE (3 runs) and parallel PSO (3 runs) for example 1.

EM simulations on average. The convergence trends of all methods are shown in Fig. 10. It can be seen that to obtain the average best objective function value of parallel SMAS (0.922nNm, 249 parallel EM simulations, 39 hours), PSAED needs 35 parallel EM simulations (5.3 hours). Hence, PSAED offers a 7.36 times speed improvement over parallel SMAS. To obtain the average best objective function values of parallel DE (0.812nNm, 987 parallel EM simulations, 6.1 days) and parallel PSO (0.913pNm, 1214 parallel EM simulations, 7.4 days), PSAED needs 17 and 29 parallel EM simulations (2.5 hours and 4.4 hours), respectively. Hence, PSAED offers a 58.6 times speed improvement over parallel DE and a 40.4 times speed improvement over parallel PSO. Therefore, PSAED shows clear advantages compared to the reference methods in terms of efficiency and optimization capacity for this example.

### B. Example 2

The second example is a hybrid DRA [41] with a layout shown in Fig. 11. The hybrid DRA is modeled in CST Microwave Studio and excited at  $TE_{\delta 11}$  mode via a slot in the ground plane. The hybrid DRA has a mesh density of 12 cells per wavelength and about 22,000 mesh cells in total. Each EM simulation costs about 30s. A moderate fidelity is used here for comparison and, in real-world design, the fidelity should be higher and each EM simulation could cost 5-10 minutes. The relative permittivity and loss tangent of the dielectric resonator (DR) are 10 and 0.0001, respectively. The substrate is RO4003C with a thickness of 0.5mm, relative permittivity of 3.38 and loss tangent of 0.0027. The substrate is placed on a copper ground of 0.05mm thickness.

As shown in (21), the design exploration goal is to minimize the maximum of reflection coefficients ( $\max(|S_{11}|)$ ) in the

Table VI  
RANGES OF THE DESIGN VARIABLES (ALL SIZES IN MM) FOR HYBRID DRA DESIGN EXPLORATION

Variables	$a_x$	$a_y$	$a_z$	$a_c$	$u_s$	$w_s$	$y_s$
Lower bound	6	12	6	6	0.5	4	2
Upper bound	10	16	10	8	4	12	12

operational band of 5.28GHz to 5.72GHz. The parameters that can be adjusted include the dimensions of the DR brick ( $a_x$ ,  $a_y$  and  $a_z$ ), the slot dimensions ( $u_s$  and  $w_s$ ), the length of the microstrip slab ( $y_s$ ) and location of the DR relative to the slot ( $a_c$ ). Their ranges are shown in Table VI. A geometric constraint of ( $a_c \leq 0.5 \times a_y$ ) is used to ensure that the slot remains under the DRA in all possible cases.

For antenna design problems, designs with  $\max(|S_{11}|) \leq -20dB$  within the bandwidth show excellent response and the smaller the better. The computing budget is as follows: 370 parallel simulations for PSAED and SMAS, respectively, and 2000 parallel simulations for DE and PSO, respectively. 10 independent runs are carried out for PSAED and SMAS, respectively, while 3 independent runs are carried out for parallel DE and PSO, respectively, because more runs are not affordable.

$$\text{minimize } \max|S_{11}| \quad 5.28GHz - 5.72GHz \quad (21)$$

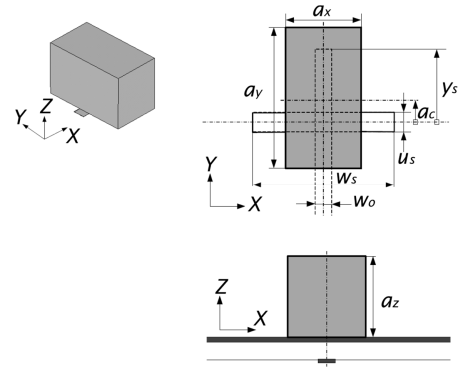


Figure 11. Layout of the hybrid DRA

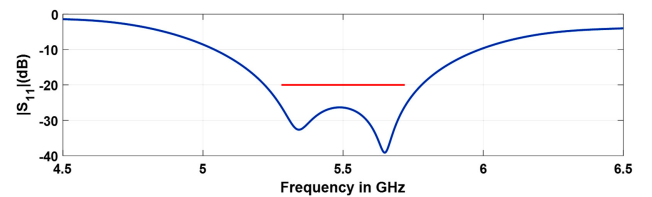


Figure 12. A typical optimized performance of the hybrid DRA by PSAED

Optimization results are shown in Table VII. A typical response of PSAED is shown in Fig. 12 with the obtained optimal design of  $a_x = 6.80mm$ ,  $a_y = 15.00mm$ ,  $a_z = 9.97mm$ ,  $a_c = 5.02mm$ ,  $u_s = 3.68mm$ ,  $w_s = 7.72mm$ ,  $y_s = 2.13mm$ . The optimization time is about 5 hours. The following observations can be made: (1) In all 10 runs, PSAED

Table VII  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES USING  
DIFFERENT METHODS (EXAMPLE 2, IN dB)

Method	Best	Worst	Mean	Median	Std.
PSAED	-26.03	-23.99	-24.58	-24.32	0.6865
Parallel SMAS	-24.03	-21.62	-23.52	-23.65	0.8957
Parallel DE	-25.03	-23.15	-24.31	-24.73	1.0116
Parallel PSO	-22.58	-19.05	-21.34	-22.40	1.9891

obtains very satisfactory results even in the worst case. (2) PSAED reveals good robustness as the standard deviation is low. (3) PSAED shows a small advantage over parallel DE, and a clear advantage over parallel SMAS and parallel PSO in terms of design solution quality.

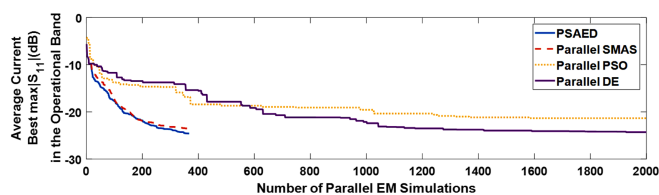


Figure 13. Convergence trends of PSAED (10 runs), parallel SMAS (10 runs), parallel DE (3 runs) and parallel PSO (3 runs) for example 2.

In terms of efficiency, PSAED uses around 360 parallel EM simulations on average. The convergence trends of all methods are shown in Fig. 13. It can be seen that to obtain the average best objective function values of parallel SMAS (-23.52dB with 364 parallel EM simulations, 5.3 hours), PSAED needs 276 EM simulations (3.9 hours). Hence, PSAED offers 1.4 times speed improvement over parallel SMAS. To obtain the average best objective function values of parallel DE (-24.31dB with 1928 parallel EM simulations, 26.8 hours) and parallel PSO (-21.34dB with 1581 parallel EM simulations, 23 hours), PSAED needs 346 and 178 parallel EM simulations (4.9 hours and 2.4 hours), respectively. Hence, PSAED offers a 5.5 times speed improvement over parallel DE, and 9.6 times over parallel PSO. Therefore, PSAED shows clear advantages compared to other alternatives in terms of both efficiency and optimization capacity for this example.

### C. Benchmark Problem Tests

In this subsection, the performance of PSAED is demonstrated using the four mathematical benchmark problems with different landscape complexity [36], [42] listed in Table VIII. More details can be found in the appendix. To make all of the functions converge, the computing budget is 100, 300, 250 and 300 exact function evaluations, respectively, for PSAED and parallel SMAS. The computing budget of parallel DE and parallel PSO is 1500 parallel evaluations on F1 - F4, respectively. A Wilcoxon test [43] is carried out using the final optimal function values obtained by all methods (over 30 independent runs) as data samples. The null hypothesis is that the data samples of PSAED and the reference methods have equal medians at 5% significance level (95% confidence

Table VIII  
MATHEMATICAL BENCHMARK PROBLEMS

Problem	Objective Function	No. of Variables	Global Optimum	Property
F1	Hartmann	6	-3.32237	Multimodal; 6 local minima
F2	Ellipsoid	10	0	Unimodal
F3	Ackley	10	0	Multimodal; Many local minima
F4	Michalewicz	10	-9.66015	Multimodal; 10! local minima; Many steeped ridges; Many valleys

Table IX  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES OBTAINED BY  
PSAED FOR F1 - F4 (OVER 30 RUNS)

Problem	Best	Worst	Mean	Median	Std.
F1	-3.3224	-3.2032	-3.2945	-3.3224	0.0513
F2	1.04e-10	3.27e-08	7.16e-09	5.11e-09	8.52e-09
F3	3.65e-04	0.0051	0.0015	0.0012	0.0012
F4	-9.3345	-8.2029	-8.8467	-8.8878	0.2724

Table X  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES OBTAINED BY  
PARALLEL SMAS FOR F1 - F4 (OVER 30 RUNS)

Problem	Best	Worst	Mean	Median	Std.	p-value
F1	-3.3223	-3.2029	-3.2824	-3.3220	0.0571	1.25e-05
F2	1.83e-05	9.00e-04	1.35e-04	6.57e-05	1.87e-04	3.02e-11
F3	0.0053	0.1696	0.0246	0.0151	0.0314	3.01e-11
F4	-9.3806	-7.0683	-8.3314	-8.3600	0.6791	0.0030

Table XI  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES OBTAINED BY  
PARALLEL DE FOR F1 - F4 (OVER 30 RUNS)

Problem	Best	Worst	Mean	Median	Std.	p-value
F1	-3.3224	-3.2032	-3.2906	-3.3224	0.0536	9.21e-05
F2	0.0910	1.2499	0.3984	0.3348	0.2929	3.02e-11
F3	0.3990	1.3489	0.8160	0.8000	0.2404	3.01e-11
F4	-6.8819	-5.3480	-6.1510	-6.1458	0.3905	3.02e-11

Table XII  
STATISTICS OF THE BEST OBJECTIVE FUNCTION VALUES OBTAINED BY  
PARALLEL PSO FOR F1 - F4 (OVER 30 RUNS)

Problem	Best	Worst	Mean	Median	Std.	p-value
F1	-3.3224	-3.2032	-3.2508	-3.2032	0.0594	0.5997
F2	9.24e-05	0.1612	0.0107	0.0018	0.0323	3.02e-11
F3	0.0017	1.6540	0.1229	0.0149	0.3596	9.89e-11
F4	-9.3639	-4.7407	-8.0584	-8.3207	1.0646	1.49e-04

Table XIII  
SPEED ENHANCEMENT OF PSAED OVER OTHER METHODS FOR F1 - F4  
(OVER 30 RUNS)

Method	F1	F2	F3	F4
Parallel SMAS	1.67	1.49	1.31	2.01
Parallel DE	17.9	11.77	9.20	23.58
Parallel PSO	19.54	7.84	11.11	11.83

level). The statistics (over 30 runs) are shown in Table IX, Table X, Table XI and Table XII.

In terms of solution quality, the following observations can be made: (1) PSAED shows excellent solution quality for F1-F3, for which the average and median values of the best function values over 30 runs are very close to the global optimum. Even for F4, which is more complex than the targeted EM design problems, the optimal result obtained by PSAED is still reasonably acceptable. (2) According to the p-values in Table X, Table XI and Table XII, the null hypothesis is rejected for all the cases except parallel PSO for F1. Hence, PSAED obtains statistically better optimization quality compared to the reference methods.

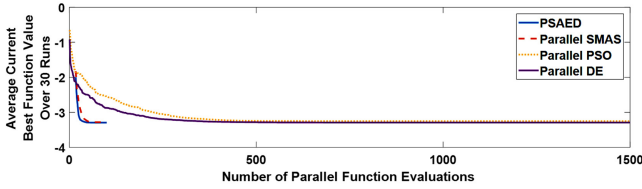


Figure 14. Convergence trends of all methods for F1.

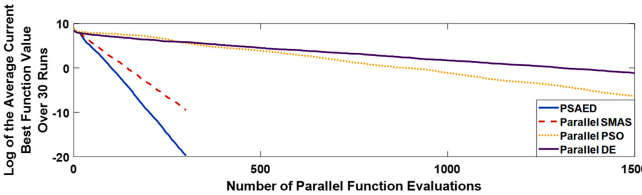


Figure 15. Convergence trends of all methods for F2.

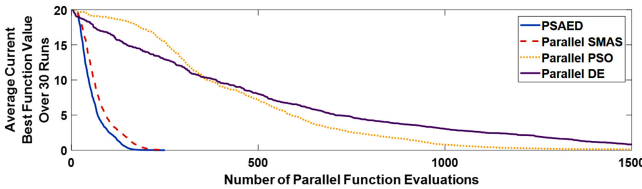


Figure 16. Convergence trends of all methods for F3.

In terms of efficiency, the convergence trends of all methods are shown in Fig. 14, Fig. 15, Fig. 16 and Fig. 17. To calculate the speed enhancement of PSAED with respect to the reference methods, the average convergence trends of all the methods are used. As in Example 1 and Example 2, the number of parallel evaluations used by PSAED to obtain the best values of the reference methods is divided by the number of evaluations

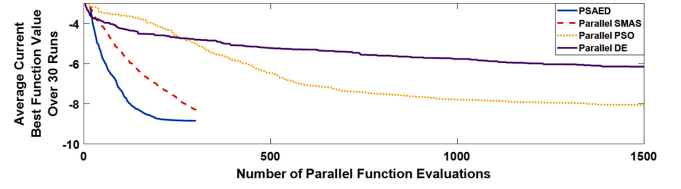


Figure 17. Convergence trends of all methods for F4.

used by the reference methods to obtain the same values. Table XIII shows this result. It can be seen that PSAED obtains from several times to up to about 20 times speed enhancement compared to parallel SMAS, parallel DE and parallel PSO.

## V. CONCLUSIONS

In this paper, the PSAED method has been proposed for efficient electromagnetic design exploration in parallel simulation environments, which is in line with the trend of cloud computing for engineering design automation. Experiments using a micromirror, a dielectric resonator antenna and four mathematical benchmark problems show that PSAED is more efficient than parallel SMAS (using the top candidate designs) and much more efficient than parallel DE and parallel PSO, while obtaining more competitive design solutions. The efficacy of the PSAED can be attributed to selecting appropriate mutation strategies for the targeted design landscape and combining them in a particular way. Future work will include developing PSAED-based electromagnetic design exploration tools and investigating the performance of PSAED in higher order parallel clusters.

## APPENDIX

### A. F1: 6-d Hartmann Problem

$$\text{minimize } F_x = -\sum_{i=1}^4 a_i \times e^{-\sum_{j=1}^d A_{ij}(x_j - P_{ij})^2}$$

$$a = (1.0, 1.2, 3.0, 3.2)^T$$

$$A = \begin{bmatrix} 10 & 3 & 17 & 3.50 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{bmatrix}$$

$$P = 10^{-4} \times \begin{bmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{bmatrix}$$

$$x \in [0, 1], i = 1, \dots, 6$$

$$\text{minimum} : x^* = (0.20169, 0.150011, 0.476874, \dots, 0.275332, 0.311652, 0.6573); f(x^*) = -3.32237$$

(22)

### B. F4: 10-d Michalewicz Problem

$$\text{minimize } F_x = -\sum_{i=1}^d \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right)$$

$$x \in [0, \pi], i = 1, \dots, 10$$

$$\text{minimum} : f(x^*) = -9.66015$$

(23)

## REFERENCES

- [1] P. Rocca, G. Oliveri, and A. Massa, "Differential evolution as applied to electromagnetics," *IEEE Antennas and Propagation Magazine*, vol. 53, no. 1, pp. 38–49, 2011.
- [2] P. Di Barba and S. Wiak, "Evolutionary computing and optimal design of MEMS," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 4, pp. 1660–1667, 2015.
- [3] P. Di Barba, M. E. Mognaschi, M. Przybylski, N. Rezaei, B. Slusarek, and S. Wiak, "Geometry optimization for a class of switched-reluctance motors: a bi-objective approach," *International Journal of Applied Electromagnetics and Mechanics*, no. Preprint, pp. 1–16, 2017.
- [4] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [5] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*. Springer, 2011, pp. 760–766.
- [6] Z. D. Zaharis, I. P. Gravas, T. V. Yioultis, P. I. Lazaridis, I. A. Glover, C. Skeberis, and T. D. Xenos, "Exponential log-periodic antenna design using improved particle swarm optimization with velocity mutation," *IEEE Transactions on Magnetics*, vol. 53, no. 6, pp. 1–4, 2017.
- [7] Z. Fan, J. Liu, T. Sorensen, and P. Wang, "Improved differential evolution based on stochastic ranking for robust layout synthesis of MEMS components," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 4, pp. 937–948, 2009.
- [8] S. Kahng, M. Khattak, H.-S. Kim, and S.-H. Oh, "A PSO-optimized, compact UWB bandpass filter suppressing the WLAN and the spurious harmonics," in *Asia-Pacific Microwave Conference*, vol. 1. IEEE, 2015, pp. 1–4.
- [9] P. Di Barba, B. Liu, M. E. Mognaschi, P. Venini, and S. Wiak, "Multiphysics field analysis and evolutionary optimization: Design of an electro-thermo-elastic microactuator," *International Journal of Applied Electromagnetics and Mechanics*, vol. 54, no. 3, pp. 433–448, 2017.
- [10] M. Mognaschi, "Micro biogeography-inspired multi-objective optimisation for industrial electromagnetic design," *Electronics Letters*, vol. 53, no. 22, pp. 1458–1460, 2017.
- [11] S. Koziel and P. Kurgan, "Rapid design of miniaturised branch-line couplers through concurrent cell optimisation and surrogate-assisted fine-tuning," *IET Microwaves, Antennas & Propagation*, vol. 9, no. 9, pp. 957–963, 2015.
- [12] G. I. Hawe and J. K. Sykalski, "A scalarizing one-stage algorithm for efficient multi-objective optimization," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 1094–1097, 2008.
- [13] I. Couckuyt, F. Declercq, T. Dhaene, H. Rogier, and L. Knockaert, "Surrogate-based infill optimization applied to electromagnetic problems," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 20, no. 5, pp. 492–501, 2010.
- [14] B. Liu, H. Yang, and M. J. Lancaster, "Global optimization of microwave filters based on a surrogate model-assisted evolutionary algorithm," *IEEE Transactions on Microwave Theory and Techniques*, vol. 65, no. 6, pp. 1976–1985, 2017.
- [15] W. Gong, A. Zhou, and Z. Cai, "A multioperator search strategy based on cheap surrogate models for evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 746–758, 2015.
- [16] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011.
- [17] B. Liu, Q. Zhang, and G. G. E. Gielen, "A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 180–192, 2014.
- [18] B. Liu, Q. Chen, Q. Zhang, G. Gielen, and V. Grout, "Behavioral study of the surrogate model-aware evolutionary search framework," in *IEEE Congress on Evolutionary Computation*. IEEE, 2014, pp. 715–722.
- [19] B. Liu, D. Zhao, P. Reynaert, and G. G. Gielen, "GASPAD: A general and efficient mm-wave integrated circuit synthesis method based on surrogate model assisted evolutionary algorithm," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 2, pp. 169–182, 2014.
- [20] B. Liu *et al.*, "An efficient method for antenna design optimization based on evolutionary computation and machine learning techniques," *IEEE Trans. on Antennas and Propagation*, vol. 62, no. 1, pp. 7–18, 2014.
- [21] B. Liu, V. Grout, and A. Nikolaeva, "Efficient global optimization of actuator based on a surrogate model assisted hybrid algorithm," *IEEE Transactions on Industrial Electronics*, vol. PP, no. 99, pp. 1–1, 2017.
- [22] B. Liu, S. Koziel, and N. Ali, "SADEA-II: A generalized method for efficient global optimization of antenna design," *Journal of Computational Design and Engineering*, vol. 4, no. 2, pp. 86–97, 2017.
- [23] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, 2006, vol. 1.
- [24] A. Bekasiewicz and S. Koziel, "Fast microwave filter optimization using adjoint sensitivities and variable-fidelity electromagnetic simulations," in *31st International Review of Progress in Applied Computational Electromagnetics*. IEEE, 2015, pp. 1–2.
- [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] 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.
- [27] M. Stein, "Large sample properties of simulations using Latin hypercube sampling," *Technometrics*, pp. 143–151, 1987.
- [28] B. Liu and A. Nikolaeva, "Efficient global optimization of MEMS based on surrogate model assisted evolutionary algorithm," in *Proceedings of Design, Automation & Test in Europe*. IEEE, 2016, pp. 555–558.
- [29] M. D. Nardo, G. L. Calzo, M. Galea, and C. Gerada, "Design optimization of a high-speed synchronous reluctance machine," *IEEE Transactions on Industry Applications*, vol. 54, no. 1, pp. 233–243, Jan 2018.
- [30] H. J. Mohammed, A. S. Abdullah, R. S. Ali, R. A. Abd-Alhameed, Y. I. Abdulraheem, and J. M. Noras, "Design of a uniplanar printed triple band-rejected ultra-wideband antenna using particle swarm optimisation and the firefly algorithm," *IET Microwaves, Antennas & Propagation*, vol. 10, no. 1, pp. 31–37, 2016.
- [31] J. L. Chávez-Hurtado and J. E. Rayas-Sánchez, "Polynomial-based surrogate modeling of RF and microwave circuits in frequency domain exploiting the multinomial theorem," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 12, pp. 4371–4381, 2016.
- [32] D. Lee, J.-W. Kim, C.-G. Lee, and S.-Y. Jung, "Variable mesh adaptive direct search algorithm applied for optimal design of electric machines based on FEA," *IEEE Transactions on Magnetics*, vol. 47, no. 10, pp. 3232–3235, 2011.
- [33] T.-Y. Lee, P. X. Trung, J.-W. Kim, Y.-J. Kim, and S.-Y. Jung, "Search region management method for local search algorithm employing design optimization of brushless DC motor," *IEEE Transactions on Magnetics*, vol. 52, no. 3, pp. 1–6, 2016.
- [34] C. Zhang, F. Feng, Q. J. Zhang, J. W. Bandler *et al.*, "Cognition-driven formulation of space mapping for equal-ripple optimization of microwave filters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 63, no. 7, pp. 2154–2165, 2015.
- [35] J. M. Dieterich and B. Hartke, "Empirical review of standard benchmark functions using evolutionary global optimization," *Applied Mathematics*, vol. 3, pp. 1552–1564, 2012.
- [36] M. Jamil *et al.*, "A literature survey of benchmark functions for global optimisation problems," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150–194, 2013.
- [37] B. Hopkins and J. G. Skellam, "A new method for determining the type of distribution of plant individuals," *Annals of Botany*, vol. 18, no. 2, pp. 213–227, 1954.
- [38] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximity-based mutation operators," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 99–119, 2011.
- [39] D. E. Goldberg and K. Deb, "A comparative analysis of selection scheme used in genetic algorithms," *Foundations of genetic algorithms*, pp. 69–93, 1991.
- [40] E. Mezura-Montes, J. Velázquez-Reyes, and C. A. Coello Coello, "A comparative study of differential evolution variants for global optimization," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 485–492.
- [41] S. Koziel and S. Ogurtsov, "Multi-point response correction for cost-efficient antenna and microwave design optimization," in *Loughborough Antennas and Propagation Conference*. IEEE, 2013, pp. 548–552.
- [42] T. Krink and R. Thomsen, "Self-organized criticality and mass extinction in evolutionary algorithms," in *Proceedings of IEEE Congress on Evolutionary Computation*, vol. 2. IEEE, 2001, pp. 1155–1161.
- [43] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.



**Mobayode O. Akinsolu (M'16)** received his M.Sc. with Distinction in Electrical and Electronic Engineering from the University of Bradford, UK in 2014, after his undergraduate studies and national service in Nigeria. He then worked as a research fellow (industrial attaché) and a visiting researcher at the National Space Research and Development Agency, Nigeria and the RFID Research Centre, Nigeria respectively. Since 2016, he has been a recipient of a PhD studentship in recognition of a joint project between Wrexham Glyndwr University,

UK and the University of Birmingham, UK. His PhD research focuses on electromagnetic design automation using surrogate model assisted evolutionary algorithms.



**Bo Liu (M'15-SM'17)** 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 University of Leuven (KU Leuven), Belgium, in 2012. From 2012 to 2013, he was a Humboldt research fellow and was working with Technical University of Dortmund, Germany. In 2013, he was appointed lecturer at Wrexham Glyndwr University, UK, where he was promoted to Reader in Computer-aided Design in 2016. He is an honorary fellow at The University of Birmingham. His research inter-

ests lie in AI-driven design methodologies of analog/RF integrated circuits, microwave devices, MEMS, evolutionary computation and machine learning. He has authored or coauthored 1 book and more than 50 papers in international journals, edited books and conference proceedings.



**Vic Grout** has a BSc(Hons) in Mathematics and Computing from the University of Exeter and a PhD in Communication Engineering from Plymouth Polytechnic. He is currently Professor of Computing Futures at Wrexham Glyndwr University, Wales; having been previously Professor of Network Algorithms, Head of Computing, Associate Dean for Research and Director of the Centre for Applied Internet Research. He is an approved British Computer Society accreditation assessor, an Institute of Engineering and Technology recommended speaker

and a European Commission 'Horizon 2020' Expert Research Assessor and Ethics consultant. Vic has worked in senior positions in academia and industry for nearly 30 years and has published over 350 research papers, patents and books. His research interests span several areas of computational mathematics, including artificial intelligence and the application of heuristic principles to large-scale problems in Internet design, modelling, simulation, management and control.



**Pavlos I. Lazaridis (M'13-SM'15)** received his B.Sc. degree in electrical engineering from Aristotle University of Thessaloniki, Greece, in 1990, his M.Sc. in electronics from Université Pierre and Marie Curie, Paris 6, France in 1992, and his Ph.D. from ENST Paris and Paris 6 in 1996. From 1991 to 1996, he was involved with research for France Telecom and teaching at ENST Paris. In 1997, he became head of the Antennas and Propagation Laboratory, TDF- C2R Metz (Telediffusion de France/France Telecom Research Center). From 1998 to 2002 he

was Senior Examiner at the European Patent Office (EPO), Den Haag, The Netherlands. From 2002 to 2014 he was involved with teaching and research at the ATEI of Thessaloniki, Greece and Brunel University. He is currently a Reader in electronic and electrical engineering at the University of Huddersfield, United Kingdom, senior member IEEE, member of the IET, and a Fellow of the Higher Education Academy.



**Maria Evelina Mognaschi (M'15)** is an assistant professor at the University of Pavia (Italy), Department of Electrical, Computer and Biomedical Engineering. Her scientific interests are inverse problems, in particular multi-objective optimization and identification problems in electromagnetism and biological systems. She has authored or co-authored more than 50 ISI- or Scopus-indexed papers, either presented to international conferences or published in international journals.



**Paolo Di Barba (M'95)** received the M.Eng. degree from the University of Pavia, Italy, in 1988, the Ph.D. degree from the Technical University of Lodz, Poland, in 2002, and the D.Sc. degree from the Lodz, University of Technology, Poland, in 2013, all in electrical engineering. He is currently a Full Professor of Electrical Engineering in the Department of Electrical, Computer, and Biomedical Engineering, University of Pavia, Pavia, Italy. His current research interests include the computer-aided design of electric and magnetic devices, with special

emphasis on the methods for field synthesis and automated optimal design. He has authored or coauthored more than 100 papers, either presented to international conferences or published in international journals, the book *Field Models in Electricity and Magnetism* (Springer, 2008) and the monograph *Multiojective Shape Design in Electricity and Magnetism* (Springer, 2010).