

Research Article

Hybrid Differential Evolution with Biogeography-Based Optimization for Design of a Reconfigurable Antenna Array with Discrete Phase Shifters

Xiangtao Li¹ and Minghao Yin^{1,2}

¹ College of Computer Science, Northeast Normal University, Changchun 130117, China

² Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China

Correspondence should be addressed to Minghao Yin, lixt314@gmail.com

Received 6 April 2011; Revised 27 August 2011; Accepted 25 September 2011

Academic Editor: Ananda Sanagavarapu Mohan

Copyright © 2011 X. Li and M. Yin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multibeam antenna arrays have important applications in communications and radar. This paper presents a new method of designing a reconfigurable antenna with quantized phase excitations using a new hybrid algorithm, called DE/BBO. The reconfigurable design problem is to find the element excitation that will result in a sector pattern main beam with low sidelobes with additional requirement that the same excitation amplitudes applied to the array with zero-phase should be in a high directivity, low sidelobe pencil-shaped main beam. In order to reduce the effect of mutual coupling between the antenna-array elements, the dynamic range ratio is minimized. Additionally, compared with the continuous realization and subsequent quantization, experimental results indicate that the performance of the discrete realization of the phase excitation value can be improved. In order to test the performances of hybrid differential evolution with biogeography-based optimization, the results of some state-of-art algorithms are considered, for the purposed of comparison. Experiment results indicate the better performance of the DE/BBO.

1. Introduction

The problem of reconfigurable antenna arrays involves radiating multiple patterns using a single power-divided network. In the past decades, this problem has been one of the most active and prolific research areas since the pioneering work of Bucci et al. [1, 2]. Moreover, this problem has also been a central and well-studied problem with a strong engineering background in the field of manufacturing and telecommunications science [3, 4]. In order to solve this problem, many methodologies have been proposed to obtain the multipattern arrays in previous literature [4–8].

Traditionally, exact algorithms such as branch and bound method and mixed-integer linear programming method have been widely used in early days to solve the problem. However, because the computational time of these methods is always unacceptable, these methods can only solve problems of relatively small size. On the other hand, evolutionary algorithms perform population-based probabilistic searches with a high speed of convergence rate and have been pro-

ved very successful in solving problems of large scale. When it comes to solving reconfigurable antenna problems, compared with traditional algorithms, evolutionary algorithms have the ability of obtaining excitation phases and amplitudes that can be practically implemented more easily by imposing additional constraints. So it is not uncommon, in the past decade, to see that different kinds of evolutionary algorithms, such as simulated annealing (SA) [4], genetic algorithm (GA) [4–6], particle swarm optimization algorithm (PSO) [7, 8], and tabu search algorithm [9], have been advanced to handle reconfigurable antenna problem, especially for the problems of large scale. Recently, two novel metaheuristic algorithms, differential evolution (DE) and biogeography based optimization (BBO), have been proposed, which have gained significant attention in the research literatures. DE is a simple yet powerful population-based, direct search algorithm with the generation and test feature for global optimization problems. The basic idea of DE is to create new candidate solutions by combining the parent individual with several other individuals of the same

population, and a candidate will replace the parent only when it has better fitness. BBO is the study of the geographical distribution of biological organisms.

In evolutionary algorithm-based antenna-array synthesis producers, phased excitations are always represented by continuous values; however, discrete phase shifters are used to realize the phase excitation sometimes. Therefore, the excitation phase values obtained by these approaches are subsequently quantized to the nearest n -bit phase shifter excitation values. In order to solve the reconfigurable antenna array with quantized phase excitations, Baskar proposed a mixed-integer optimization for the first time in an evolution search method, namely, the generalized generation gap model GA (G3-GA) [10]. The objective is to optimize real-valued amplitude excitations and quantized phase excitations [11].

Although meta-heuristic methods have been proved to have superior features to other traditional methods, they also suffer some limitations. The performance of the conventional DE algorithm highly depends on the mutation and crossover operator. In some cases, they may be easy to fall into the local minimum or converge too slowly. Being fascinated by the prospect and potential of DE, recently, many researches are working on the improvement of DE, and many variants of the new algorithm are presented. Brest et al. [12] proposed a self-adaptive parameter setting in differential evolution in order to avoid the manual parameter setting of F and CR. The parameter control technique is based on the self-adaptation of two parameters associated with the evolutionary process. The main goal is to produce a flexible DE, in terms of control parameters F and CR. The result shows that the algorithm with self-adaptive control parameter setting is better than, or at least comparable to, the standard DE algorithm and evolutionary algorithms from the literature. Rahnamayan et al. [13] proposed an opposition-based differential evolution, called ODE. The ODE algorithm consisted of a DE framework and two opposition-based components: the former after the initial sampling and the latter after the survivor selection scheme. Liu and Lampinen [14] introduced a new version of the differential evolution algorithm with control parameters. The algorithm uses the fuzzy logic controllers to adapt the parameters. Qin et al. [15] proposed a self-adaptive DE algorithm (SaDE), in which both trail vector generation strategies and the associated control parameter values were gradually self-adaptive by learning from their previous experiences when generating promising solutions. This method did not use any particular learning strategy, nor any specific setting for the control parameters F and CR. This algorithm was more effective in obtaining better quality solution, which are more stable with the relatively smaller standard deviation and have a higher success rate.

Additionally, researchers have found that a skilled combination of two meta-heuristic techniques can improve the performance of the algorithms obviously when dealing with real-world and large-scale problems [16]. Some hybrid-heuristic-based optimization methods have been investigated in the past few years [17–19]. Sun et al. [20] proposed a combination of DE algorithms and the estimation of distribution algorithm (EDA), in which new promising solutions are

created by DE/EDA. This algorithm uses a probability model to determine promising regions in order to focus the search process on those areas. Norman and Iba [21] proposed a crossover-based adaptive local search operation for enhancing the performance of standard differential evolution algorithm; this algorithm combines the DE with fittest individual refinement (FIR). The FIR scheme accelerates DE by applying a fixed-length crossover-based in the neighborhood of the best solution in each generation. Gong et al. [22] proposed an improved version of DE, namely, orthogonal based DE. This algorithm employed the two-level orthogonal crossovers to improve the performance of DE. Omran et al. [23] proposed the barebones differential evolution that is a hybrid of the barebones particle swarm optimizer and differential evolution. DE is used to mutate for each particle, and the attractor is associated with that particle, defined as a weighted average of its personal and neighborhood best positions. Neri and Tirronen [24] proposed the scale factor local search differential evolution. This algorithm employs, within a self-adaptive scheme, two local search algorithms. These local search algorithms aim at detecting a value of the scale factor corresponding to an offspring with a high performance. A statistical analysis of the optimization results has been included in order to compare the results in terms of final solution detected and convergence speed. Yang et al. [25] proposed the neighborhood search differential evolution. In this algorithm, the scale factor is adjusted by the sampling value according to probability distributions, and the mutation is updated by a logic inspired by evolutionary programming.

Although, this field of study is still in its early days, a large number of future researches are necessary in order to develop hybrid algorithms for optimization problems. In [26], Gong et al. have proposed a new algorithm, called DE/BBO. This algorithm combines the exploration of DE with the exploitation of BBO effectively, and hence it can generate the promising candidate solutions. This algorithm has been used better in many areas, such as global number optimization, emission load dispatch problem, and economic emission load dispatch problem.

In this paper, we will use the hybrid differential evolution algorithm with biogeography-based optimization to solve reconfigurable antenna array optimization with quantized phase excitations. Specifically, a hybrid migration operator of BBO is adopted, combined with mutation, crossover, and selection operators of DE to explore and exploit the search space effectively. In order to demonstrate the advantages of the proposed design, the results obtained using continuous-phase excitations followed by quantization are compared.

The rest of this paper is organized as follows: in Section 2 we will introduce the problem formulation. Section 3 describes the fitness function. Section 4 describes the differential evolution. Section 5 describes the biogeography-based optimization. Section 6 describes the hybrid differential evolution and biogeography-based optimization. Corresponding experimental results are given in Section 7. In the last section we conclude this paper and point out some future research directions.

TABLE 1: Design Specifications.

Design parameters	Pencil pattern	Sector pattern
Side-lobe level (SLL)	-30 dB	-25 dB
Half-power bandwidth (HPBW)	6.8°	24°
Bandwidth at SLL	20°	40°
Ripple	NA	0.5 dB

2. Problem Formulation

The problem described is as follows: in order to design a reconfigurable dual-beam antenna array, an amplitude distribution can generate either a pencil-shaped or a sector power pattern, when the phase distribution of the array is modified appropriately. All excitation phases are set at 0° for the pencil-shaped beam and varied in the range $-180^\circ \leq \phi \leq 180^\circ$ for the sector pattern [7]. If the excitation is symmetrical about the centre of the linear array, the array with even number of uniformly spaced isotropic elements ($2N$) can be written as [10]

$$F(\theta) = 2 \sum_{k=1}^N (a_{kR} \cos \phi_k - a_{kI} \sin \phi_k) \quad (1)$$

with

$$\phi_k = \frac{2\pi}{\lambda} d_k \sin \theta, \quad (2)$$

where d_k is the distance between the position of the k th element and the centre, θ is the scanning angle from broadside, a_{kR} is the real parts of the k th element excitation, a_{kI} is the imaginary parts of the k th element excitation, and a_{kR} and a_{kI} are stotted within the range $(0, 1)$ and $(-1, 1)$, respectively. N excitation amplitude and phase coefficients are chosen to optimize the desired pattern. The pencil and sector patterns should have a high directivity; low sidelobe pencil-shaped main beam and a wide-sector beam.

3. Fitness Function Evaluation

For the reconfigurable dual-beam optimization, the objective of the fitness function must qualify the entire array radiation pattern. The calculated pattern can be described in terms of the criteria of the desired pattern. The fitness function for the dual-beam optimization can be described as follows [7]:

$$E(P) = \sum_{i=1}^3 (P_{i,d}^{(p)} - P_i^{(i)})^2 + \sum_{i=1}^4 (P_{i,d}^{(s)} - P_i^{(s)})^2, \quad (3)$$

where the superscript p is the design specification for the pencil pattern, the superscript s is the design specification of the sector pattern, the superscript d indicates the desired value of the design specification, and P indicates the applicable fitness factor in Table 1. The first part of this fitness function is summarized in the first column of Table 1, and the other part of this function is summarized in the second column. Different from the fitness function of the pencil beam pattern, the sector pattern needs to calculate the pattern ripple.

In order to decrease the effect of coupling between elements, an additional term is included in the objective function equation (4) [10]. The ratio is used to minimize the coupling effect between the maximum and minimum excitation amplitudes. The minimization of the amplitude-excitation dynamic range (ARD) can reduce the mutual coupling problem [18, 19]. The objective function can be expressed as follows:

$$Ec(P) = \sum_{i=1}^3 (P_{i,d}^{(p)} - P_i^{(i)})^2 + \sum_{i=1}^4 (P_{i,d}^{(s)} - P_i^{(s)})^2 + \text{ADR}, \quad (4)$$

where ADR is the amplitude-dynamic ratio. The ADR is defined as the ratio between the maximum excitation amplitude to the minimum excitation amplitude. The differences between the excitation amplitudes are minimized by minimizing the ADR; therefore, the effect of coupling can be minimized.

4. Differential Evolution Algorithm

Differential Evolution (DE) is an evolutionary algorithm first introduced by Storn and Price [27]. Similar to other evolutionary algorithms particularly genetic algorithm, DE uses some evolutionary operators like selection recombination and mutation operators. Different from genetic algorithm, DE uses distance and direction information from current population to guide the search process. The crucial idea behind DE is a scheme for producing trial vectors according to the manipulation of target vector and difference vector. If the trial vector yields a lower fitness than a predetermined population member, the new trial vector will be accepted and be compared in the following generation. Different kinds of strategies of DE have been proposed based on the target vector selected and the number of difference vectors used. In this paper, we use two strategies, DE/rand/1/bin, described as follows.

For each target vector $x_i(t)$, trial vector $v_i(t)$, $i = 1, \dots, \text{NP}$, let N be the dimension of target vector and G the G generation. The mutant vectors are generated in these DE/rand/1/bin strategies, respectively,

for DE/rand/1/bin

$$v_{i,G} = x_{a,G} + F(x_{b,G} - x_{c,G}), \quad (5)$$

where $a, b, c, d \in [1, \dots, \text{NP}]$ are randomly chosen integers and $a \neq b \neq c \neq d \neq i$. F is the scaling factor controlling the amplification of the differential evolution.

The cross-over operator implements a recombination of the trial vector and the parent vector to produce offspring. This operator is calculated as

$$u_{j,i,G} = \begin{cases} v_{j,i,G}, & (\text{rand}_j[0, 1] \leq \text{CR}) \text{ or } (j = j_{\text{rand}}), \\ x_{j,i,G}, & \text{otherwise,} \end{cases} \quad (6)$$

where $j = [1, \dots, D]$, $\text{rand}_j \in [0, 1]$, $j_{\text{rand}} = [1, \dots, D]$ is the randomly chosen index, CR is the crossover rate, $v_{j,i,G}$ is the

```

procedure Algorithm description of DE algorithm
begin
Step 1: Set the generation counter  $G = 0$ ; and randomly initialize a
population of NP individuals  $X_i$ . Initialize the parameter  $F$ , CR
Step 2: Evaluate the fitness for each individual in  $P$ 
Step 3: while stopping criteria is not satisfied do
  for  $i = 1$  to NP
    select randomly  $a \neq b \neq c \neq d \neq i$ 
    for  $j = 1$  to  $D$ 
       $j_{\text{rand}} = \lfloor \text{rand}(0, 1) * D \rfloor$ 
      If  $\text{rand}(0, 1) \leq \text{CR}$  or  $j == j_{\text{rand}}$  then
         $u_{i,j} = x_{a,j} + F \times (x_{b,j} - x_{c,j})$ 
      Else
         $u_{i,j} = x_{i,j}$ 
      end if
    end for
  end for
  for  $i = 1$  to NP do
    Evaluate the offspring  $u_i$ 
    If  $u_i$  is better than  $P_i$  then
       $P_i = u_i$ 
    end if
  end for
  Memorize the best solution achieved so far
Step 4: end while
end

```

ALGORITHM 1

difference vector of the j th particle in the i th dimension at the G th iteration, and $u_{j,i,G}$ denotes the trail vector of the j th particle in the i th dimension at the G th iteration. Selection operator is used to choose the next population between the trail population and the target population:

$$x_{i,G+1} = \begin{cases} u_{i,G}, & f(u_{i,G}) < f(x_{i,G}), \\ x_{i,G}, & \text{otherwise.} \end{cases} \quad (7)$$

The standard differential evolution algorithm can be described as in Algorithm 1.

5. Biogeography-Based Optimization

Biogeography-based optimization [28] is a new evolutionary algorithm developed for the global optimization. It is inspired by the immigration and emigration of species between islands in search of more friendly habitats. Each solution is called a ‘‘habitat’’ with a habitat suitability index (HSI) and represented by an n -dimension real vector. The initial individual of the habitat vectors is randomly generated. Those solutions that are good are considered to be habitats with a high HSI. Those that are poor are considered to be habitats with a low HSI. The high HSI tends to share their features with low HSI. Low HSI solutions accept a lot of new features from high HSI solutions. In BBO, a habitat H is a vector of N (SIVs) initialized randomly and then follows migration and mutation step to reach the optimal solution. The new candidate habitat is generated from all of the solutions in population by using the migration and mutation operators.

In BBO, the migration strategy is similar to the evolutionary strategy in which many parents can contribute to a single offspring. BBO migration is used to change existing solution and modify existing island. Migration is a probabilistic operator that adjusts a habitat X_i . The probability X_i is modified proportional to its immigration rate λ_i , and the source of the modified probability comes from X_i which is proportional to the emigration rate μ_j . The immigration rate and the emigration rate are functions of the number of species in the habitat. They can be calculated as follows:

$$\begin{aligned} \lambda_i &= I \left(1 - \frac{i}{n} \right), \\ \mu_j &= E \left(\frac{j}{n} \right), \end{aligned} \quad (8)$$

where I is the maximum possible immigration rate, E is the maximum possible emigration rate, i is the number of species of the i th individual, and n is the maximum number of species. Migration can be described as in Algorithm 2.

Mutation is a probabilistic operator that randomly modifies habitat SIVs based on the habitat’s a priori probability of existence. Very high HSI solutions and very low HSI solutions are equally improbable. Medium HSI solutions are relatively probable. The mutation rate m is expressed as

$$m = m_{\text{max}} \left(\frac{1 - P_s}{P_{\text{max}}} \right), \quad (9)$$

```

procedure Habitat migration
Begin
  for  $i = 1$  to NP
    Select  $X_i$  with probability based on  $\lambda_i$ 
    if  $\text{rand}(0, 1) < \lambda_i$  then
      for  $j = 1$  to NP
        Select  $X_j$  with probability based on  $\mu_j$ 
        if  $\text{rand}(0, 1) < \mu_j$  then
          Randomly select an SIV  $\sigma$  from  $X_j$ 
          Replace a random SIV in  $X_i$  with  $\sigma$ 
        end if
      end for
    end if
  end for
End

```

ALGORITHM 2

```

procedure Mutation
Begin
  for  $i = 1$  to NP
    Compute the probability  $P_i$ 
    Select SIV  $X_i(j)$  with probability based on  $P_i$ 
    if  $\text{rand}(0, 1) < m_i$  then
      Replace  $X_i(j)$  with a randomly generated SIV
    end if
  end for
End

```

ALGORITHM 3

where m_{\max} is a user-defined parameter. This mutation scheme tends to increase the diversity among the population. Mutation can be described as in Algorithm 3.

The basic structure of BBO algorithm can be described with Algorithm 4.

6. DE/BBO Approach

In this section, different steps of DE/BBO approach are described as follows.

6.1. Hybrid Migration Operation. The crucial operator of DE/BBO is the hybrid migration operator, which hybridizes the DE operator with the migration of BBO. In this algorithm, we can find that the child population U_i takes new features from different sides, including mutation operation of DE, migration operation of BBO, and corresponding parents of X_i of offspring. The core idea of the proposed hybrid migration operator is based on two considerations. First, poor solutions can accept a lot of new features from good solutions. Second, the mutation operator of DE is able to explore the new search space. Pseudocode of hybrid migration operation is available in Algorithm 3 of [26]. It is worth pointing out that in this algorithm the “DE/rand/1” mutation operator is illustrated (Algorithm 5).

6.2. Main Procedure of DE/BBO. By incorporating the above-mentioned hybrid migration operator into DE, the DE/BBO has been developed as a new algorithm. Hybrid method is described as in Algorithm 6.

As we all know, the standard DE algorithm is good at exploring the search space and locating the region of global minimum, but it is relatively slow at exploitation of the solution. On the other hand, standard BBO algorithm is usually quick at the exploitation of the solution though its exploration ability is relatively poor. Therefore, in this literature, a hybrid meta-heuristic algorithm by integrating biogeography-based optimization into differential evolution, called DE/BBO, is used to solve the problem of reconfigurable antenna array. The difference between DE/BBO and DE is that the hybrid migration operator is used to replace the original DE mutation operator. In this way, this method can explore the new search space by the mutation of the DE algorithm and exploit the population information with the migration of BBO and therefore can overcome the lack of the exploitation of the DE algorithm.

7. Experimental Results

To evaluate the performance of the DE/BBO, two experiments are conducted in this paper. The benchmark problems for the experiments are also used in [10]. In the first experiment, for each instance, there are twenty design parameters with continuous values. In the second experiment, for each instance, there are also twenty design parameters. Among them, ten phase coefficients are represented as discrete variables, and the other ten are represented as continuous variables.

In experiment I, the results of the excitation phases cannot be usable and approximate to the nearest values for an n -bit phase. In this paper, we will compare DE/BBO with differential evolution algorithm and generalized generation gap GA (G3-GA) based on these two experiments. In experiment II, 10-phase excitations are indicated as quantized values corresponding to the n -bit phase shifter used. Therefore, the values of the phase excitation are quantized between -180° and 180° with 5.625° per step. For simulating DE/BBO, the population size NP is 20. The maximum function evaluations are 20000. The crossover rate CR is $0.9 + \text{rand} * 0.05$. The scale factor F is $0.5 + \text{rand} * 0.01$. For simulating differential evolution algorithm and generalized generation gap GA (G3-GA), the population size NP is 20, the maximum function evaluations are 20000, the crossover rate CR is 0.9, and the scale factor F is 0.5. In G3-GA, the number of the offspring $\lambda = 6$, the maximum function evaluations are 20000, the population size NP is 500, and $\sigma_\alpha = \sigma_\beta = 0.25$. In order to compare fairly, we set these algorithms the same fitness evaluations.

7.1. Optimization without ARD. In this section, we will use DE/BBO for the reconfigurable antenna-array design without the coupling effects using the objective function (3). Table 2 shows the results of the excitation amplitude and phase. The best of optimal results for the experiment I, experiment I (after quantization of the phase excitations),

```

procedure Biogeography-based optimization
Begin
  Initialize the population  $P$  randomly and each habitat corresponding to a potential solution to the given problem
  Evaluate the fitness for each individual in  $P$ 
   $G = 1$ 
  While the termination criteria is not satisfied do
    Sort the population from best to worst
    For each habitat, map the HSI to the number of species  $S$ , the immigration rate  $\lambda$ , and the emigration rate  $\mu$ 
    Probabilistically use Immigration Island based on the immigration rates
    Modify the population with the migration operator shown in Habitat migration
    Update the probability for each individual
    Mutate the population with the mutation operation
    Evaluate the fitness for each individual in  $P$ 
    Sort the population from best to worst
     $G = G + 1$ 
  end while
End

```

ALGORITHM 4

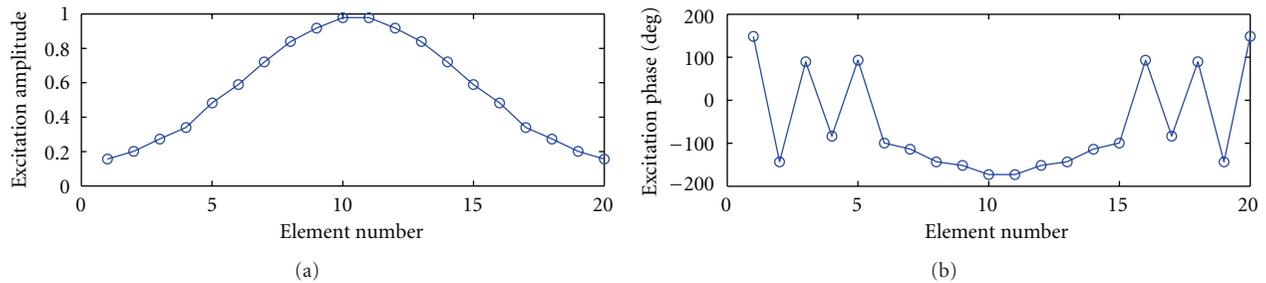


FIGURE 1: Amplitude and phase excitation (experiment I).

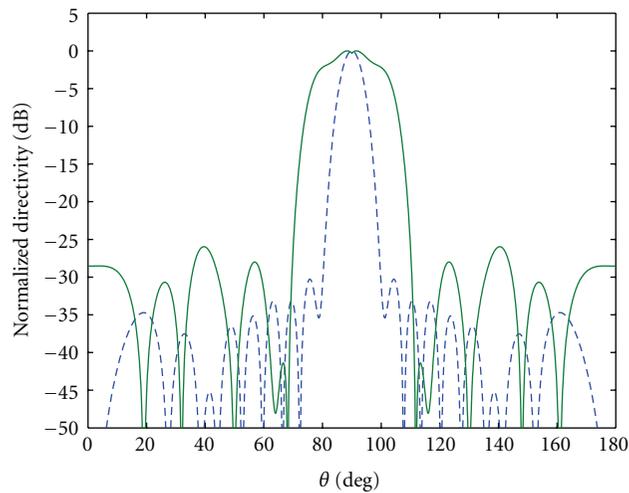


FIGURE 2: Dual-beam array patterns (experiment I).

and experiment II are listed in Table 2. The table also illustrates the ADR of the optimized excitation amplitudes and fitness function value. The optimized excitation patterns and dual-beam patterns are shown in Figures 1 and 2, respectively. Figure 2 illustrates the satisfaction of designed parameters simultaneously for both pencil and sector beam.

For the DE/BBO, the best dual-beam pattern is 0.16 for experimental I. From Table 2, we can find that the fitness value increases to 3.40 after quantizing the optimum phase values to the nearest 6-bit phase-shifter values because the sector beam increases most of the fitness value. From the quantization of the optimum result obtained in experiment

TABLE 2: Optimum results of experimental I and experiment II without ADR.

Element number	Experiment I		Experiment I after quantization		Expt-II	
	Amplitude	Phase (deg.)	Amplitude	Phase (deg.)	Amplitude	Phase (deg.)
1/20	0.1563	-172.8693	0.1563	-174.2857	0.1457	-2.8571
2/19	0.2013	-151.6404	0.2013	-151.4286	0.1697	-37.1429
3/18	0.2730	-143.0201	0.2730	-140.0000	0.2008	-31.4286
4/17	0.3396	-113.6656	0.3396	-111.4286	0.2948	-71.4286
5/16	0.4825	-99.4233	0.4825	-100.0000	0.4208	-77.1429
6/15	0.5887	93.2511	0.5887	100.0000	0.4794	-94.2857
7/14	0.7205	-83.1503	0.7205	-82.8571	0.6518	-88.5714
8/13	0.8392	89.3217	0.8392	94.2857	0.7023	-94.2857
9/12	0.9180	-143.0237	0.9180	-140.0000	0.7922	-37.1429
10/11	0.9785	148.3569	0.9785	151.4286	0.8156	31.4286
ADR		6.26		5.77		5.60
Fitness value		0.16		3.40		0.16

TABLE 3: Effects of quantization on different design specifications.

		Pencil beam			Sector beam			Ripple	Fitness
		HPBW	SLLBW	SLL	HPBW	SLLBW	SLL		
DE	Continuous phase excitation (Expt-I)	0.4	0	0	0	0	0	0	0.16
	After quantization	0.4	0	0	1.8	1.2	0	0	4.84
	Optimization with discrete variable (II)	0.6	0	0	0	0	0	0	0.36
DE/BBO	Continuous phase excitation (Expt-I)	0.4	0	0	0	0	0	0	0.16
	After quantization	0.4	0	0	0	1.4	1.3	0	3.40
	Optimization with discrete variable (II)	0.4	0	0	0	0	0	0	0.16

I may not be optimum for the discrete case. Hence, in the evolutionary process, discrete values represent the phase excitation which can eliminate the error arising due to quantization.

Table 3 illustrates the deviation between the desired and the computed design specification of the optimized results in experiment I and experiment II. Compared with the quantization of the phase values in continuous formulation, we can find that the direct discrete phase excitation formulation can generate a better fitness value. Figure 3 shows experiment I (after quantization) and experiment II for the dual-beam patterns. Difference between experiment I and experimental II is clearly shown. The best amplitude and phase excitations with discrete values are given in Figure 4.

7.2. Optimization with ADR. In this section, we will use the DE/BBO for the reconfigurable antenna-array design with the coupling effects using the objective function (4). Table 4 lists the results of experiment I and experiment II. The table also generates the ADR and the fitness values. The best fitness is less than the previous in this case. Furthermore, in experiment I, the ADR is reduced from 6.26 to 4.28. In experiment II, the ADR is reduced from 5.6 to 4.27.

Therefore, we can reduce the coupling effects by minimizing the dynamic range ratio. Figures 5 and 6 show the excitation pattern and dual-beam pattern obtained in experiment I. Figures 7 and 8 show the excitation pattern and dual-beam pattern obtained in experiment II.

7.3. Comparison of DE/BBO with DE and G3-GA [10]. In order to study the effect of the DE/BBO, we carried out a scalability study to compare the algorithm with the generalized generation gap genetic algorithm and differential evolution. The experiment is conducted for the determination of amplitude and phase excitation patterns for the dual-beam optimization with quantization. The best fitness is reported in Table 5. From Table 5, we can find that the DE/BBO can obtain better solution for experiment I and experiment II. In particular, for the dual-beam optimization with quantization, DE/BBO can perform better than G3-GA and DE. The DE/BBO can obtain the value of 0.16 better than the G3-GA's value 0.618 and the DE's value 0.36. By minimizing the dynamic ratio, we can find that the DE/BBO can provide the 4.27 (ADR) and 0.08 (fitness) better than those of G3-GA and DE. This demonstrated that DE/BBO is well suitable to solve the dual-beam optimization problem.

TABLE 4: Optimum results of Experimental I and Experiment II with ADR.

Element Number	Experiment I		Experiment I after quantization		Expt-II	
	Amplitude	Phase (deg.)	Amplitude	Phase (deg.)	Amplitude	Phase (deg.)
1/20	0.2282	-12.9749	0.2282	-8.5714	0.2240	180.0000
2/19	0.2282	-24.8762	0.2282	-20.0000	0.2240	-151.4286
3/18	0.2515	-47.4456	0.2515	-42.8571	0.2421	180.0000
4/17	0.3693	-66.2569	0.3693	-65.7143	0.3678	-88.5714
5/16	0.5090	-77.2805	0.5090	-77.1429	0.4881	77.1429
6/15	0.5880	-100.4655	0.5880	-100.0000	0.5853	71.4286
7/14	0.7815	105.7039	0.7815	111.4286	0.7532	-105.7143
8/13	0.8253	-23.9289	0.8253	-20.0000	0.8153	94.2857
9/12	0.9633	19.5264	0.9633	20.0000	0.9352	-122.8571
10/11	0.9771	-89.8865	0.9771	-88.5714	0.9568	128.5714
ADR		4.28		4.35		4.27
Fitness value		0.08		19.9099		0.09

TABLE 5: Comparison of G3-GA with DE.

	Exp-I without ADR	Exp-II without ADR	Exp-I with ADR		Exp-II with ADR	
	Fitness	Fitness	ADR	Fitness	ADR	Fitness
G3-GA	0.16	0.619	4.4137	0.1028	5.8026	0.2630
DE	0.16	0.36	4.3470	0.04	4.7190	0.16
DE/BBO	0.16	0.16	4.28	0.08	4.27	0.08

procedure Hybrid migration-operator of DE/BBO
Begin

```

for  $i = 1$  to NP
  select randomly  $a \neq b \neq c \neq d \neq i$ 
   $j_{\text{rand}} = \lfloor \text{rand}(0,1) * D \rfloor$ 
  for  $j = 1$  to  $D$ 
    if  $\text{rand}(0,1) < \lambda_i$  then
      if  $\text{rand}(0,1) \leq \text{CR}$  or  $j = j_{\text{rand}}$  then
         $u_{i,j} = x_{a,j} + F \times (x_{b,j} - x_{c,j})$ 
      else
        Select  $x_k$  with probability based on  $\mu_k$ 
         $u_{i,j} = x_{k,j}$ 
      end if
    else
       $u_{i,j} = x_{i,j}$ 
    end if
  end for
end for
End

```

ALGORITHM 5

7.4. *Effects of the Parameter.* CR = 0.9 has been widely used in most DE-related papers. Moreover, this has been proved to be the best values in standard DE in previous literatures [13, 14, 27, 29]. In order to further verify that CR = 0.9 is the best value for the problem in our literature, we have conducted two experiments. In the first experiment, the values of 20 design parameters are continuous. And in the next

experiment, 10 phase coefficients are represented by discrete variables, while the other 10 are represented by continuous variables. The experimental results can be seen in Table 6. As it can be seen in this table, in both experiments, the proposed algorithm can generate the best solutions when CR = 0.9.

7.5. *Comparison of DE/BBO with DE, CoDE, SaDE, jDE, and JADE.* In order to evaluate the effectiveness and efficiency of DE/BBO, we compare its performance with DE, CoDE [30], SaDE [15], jDE [12], and JADE [31]. Brest et al. [12] proposed a self-adaptive parameter setting in differential evolution in order to avoid the manual parameter setting of F and CR. The parameter control technique is based on the self-adaption of two parameters associated with the evolutionary process. Qin et al. [15] proposed a self-adaptive DE algorithm (SaDE), in which both trail vector generation strategies and their associated control parameter values are gradually self-adaptive by learning from their previous experiences in generating promising solutions. In the JADE proposed by Zhang and Sanderson [31], a normal distribution and a Cauchy distribution are utilized to generate F and CR for each target vector, respectively. JADE extracts information from the recent successful F and CR and uses such information for generating new F and CR. Wang et al. [30] proposed a novel method, called composite DE (CoDE), which has been proposed in this paper. This method uses three trial vector generation strategies and three control parameter settings. It randomly combines them with generate trial vectors. Each method was run 30 times on each test function. Table 7 summarizes the experimental results. As can be seen

```

Procedure Algorithm description of DE/BBO algorithm
Begin
Step 1: Set the generation counter  $G=0$ ; and randomly initialize a population of NP
individuals  $X_i$ . Initialize the parameter  $F$ , CR
Step 2: Evaluate the fitness for each individual in  $P$ 
Step 3: while stopping criteria is not satisfied do
  For each individual calculate species count probability
  Calculate the immigration rate  $\lambda_i$  and the emigration rate  $\mu_k$  for each individual
  for  $i = 1$  to NP
    select randomly  $a \neq b \neq c \neq d \neq i$ 
     $j_{\text{rand}} = \lfloor \text{rand}(0,1) * D \rfloor$ 
    for  $j = 1$  to  $D$ 
      if  $\text{rand}(0,1) < \lambda_i$  then
        if  $\text{rand}(0,1) \leq \text{CR}$  or  $j = j_{\text{rand}}$  then
           $u_{i,j} = x_{a,j} + F \times (x_{b,j} - x_{c,j})$ 
        else
          Select  $x_k$  with probability based on  $\mu_k$ 
           $u_{i,j} = x_{k,j}$ 
        end if
        else
           $u_{i,j} = x_{i,j}$ 
        end if
      end for
    end for
    for  $i = 1$  to NP do
      Evaluate the offspring  $u_i$ 
      If  $u_i$  is better than  $P_i$  then
         $P_i = u_i$ 
      end if
    end for
    Memorize the best solution achieved so far
  Step 4: end while
End

```

ALGORITHM 6

TABLE 6: Effect of the parameter of CR.

	Best	Exp-I without ADR	Exp-II without ADR	Exp-I with ADR		Exp-II with ADR	
		Fitness	Fitness	Fitness	ADR	Fitness	ADR
CR = 0.1	DE	0.79	0.40	6.47	0.42	5.77	1.09
	DE/BBO	0.45	0.82	7.38	0.06	5.43	0.33
CR = 0.3	DE	0.72	0.84	5.77	0.96	5.80	0.04
	DE/BBO	0.36	0.36	4.32	0.11	4.40	0.17
CR = 0.5	DE	0.36	0.36	4.41	0.23	4.57	0.30
	DE/BBO	0.16	0.36	4.33	0.10	4.32	0.07
CR = 0.7	DE	0.36	0.36	4.28	0.12	4.55	0.22
	DE/BBO	0.16	0.16	4.31	0.09	4.35	0.09
CR = 0.9	DE	0.16	0.36	4.3470	0.04	4.7190	0.16
	DE/BBO	0.16	0.16	4.28	0.08	4.27	0.08

in Table 6, DE/BBO significantly outperforms DE, CoDE [30], SaDE [15], jDE [12], and JADE [31] for experiment I and experimental II. By minimizing the dynamic ratio, we can find that the DE/BBO can provide the 4.27 (ARD) and 0.08 (fitness) better than those of the other algorithms.

7.6. Computational Running Time. In this section, we investigate the computational running time and the computational time complexity of the proposed algorithm. For each test function, the average running time on the 30 runs is recorded. The computational conditions are listed as follows.

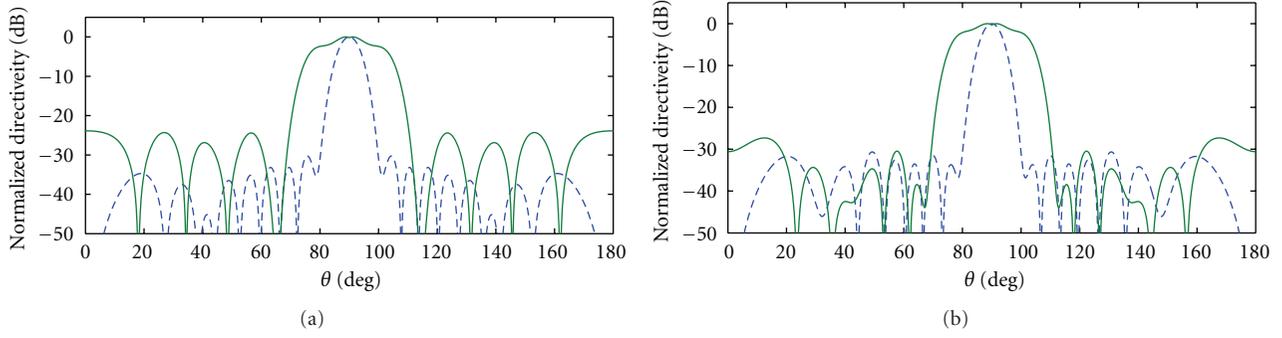


FIGURE 3: Dual-beam array pattern: experiment I (top) after quantization and experiment II.

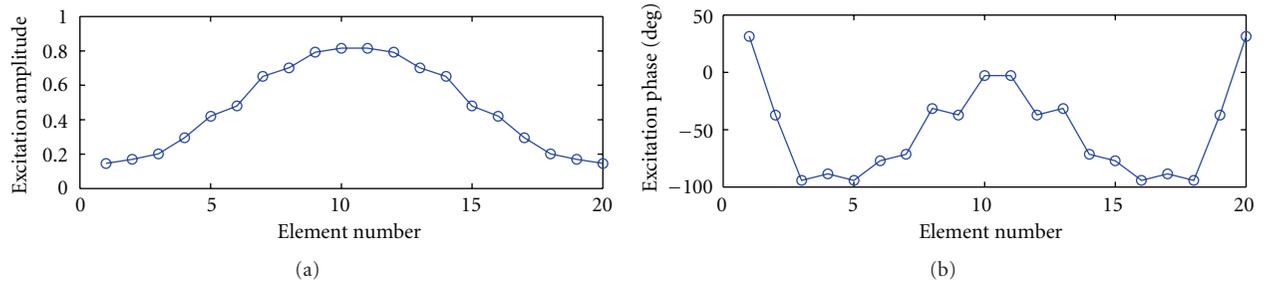


FIGURE 4: Amplitude and phase excitation (experiment II).

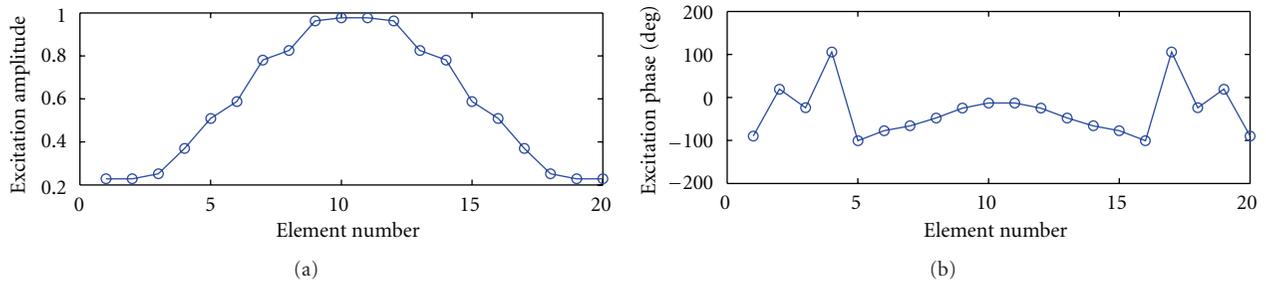


FIGURE 5: Amplitude and phase excitation (experiment I) with coupling effect.

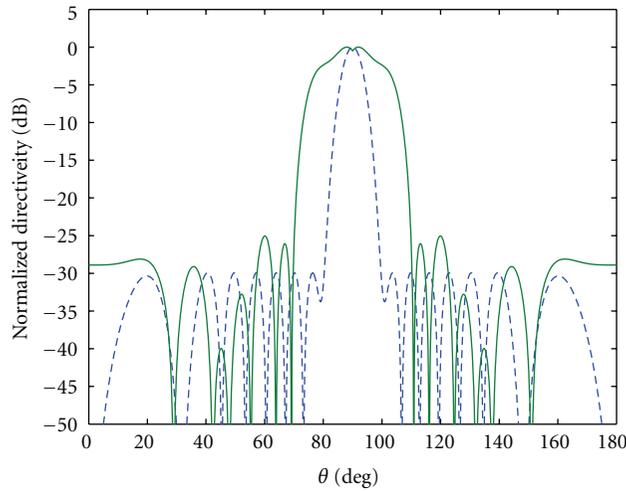


FIGURE 6: Dual-beam array pattern (experiment I) with coupling effect.

TABLE 7: Comparison of DE/BBO with DE, CoDE, SaDE, jDE, JADE.

	Exp-I without ADR	Exp-II without ADR	Exp-I with ADR		Exp-II with ADR	
	Fitness	Fitness	ADR	Fitness	ADR	Fitness
DE	0.16	0.36	4.3470	0.04	4.7190	0.16
CoDE	0.64	0.64	4.94	0.23	4.31	0.09
SaDE	0.16	0.36	4.63	0.28	4.38	0.12
jDE	0.36	0.36	4.40	0.10	4.33	0.26
JADE	0.36	0.17	4.35	0.09	4.47	0.16
DE/BBO	0.16	0.16	4.28	0.08	4.27	0.08

TABLE 8: The average computation running time (in seconds) achieved by different algorithm.

Average time (sec)	Exp-I without ADR	Exp-II without ADR	Exp-I with ADR	Exp-II with ADR
DE	252.765	261.093	256.094	256.98
CoDE	259.437	180.5470	252.422	172.844
SaDE	252.356	253.126	257.362	253.153
jDE	254.406	252.438	255.016	254.4840
JADE	279.375	265.812	251.331	268.234
DE/BBO	256.500	248.500	251.250	249.531

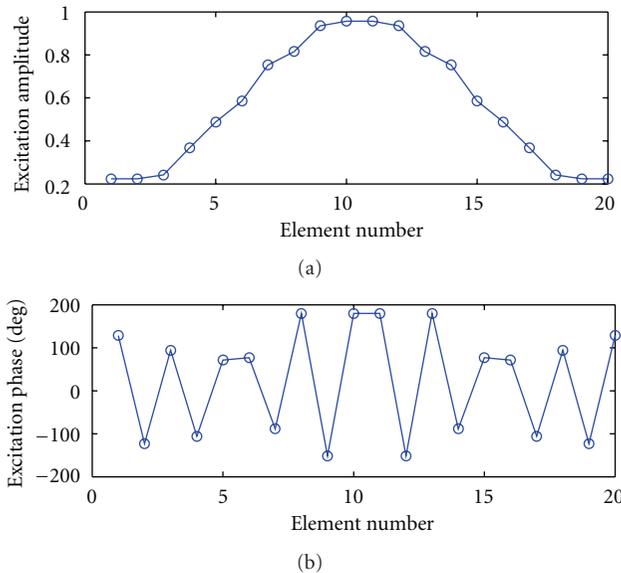


FIGURE 7: Amplitude and phase excitation (experiment II) with coupling effect.

System: Windows XP

(i) CPU: Intel(R) Core(TM) 2 Quad.

(ii) RAM: 1G.

(iii) Language: Matlab.

(iv) Compiler: Matlab 7.0.

(v) MAX FEs: 20000 ($D = 20$ and popsize = 20).

The average computation running time of different algorithms on the test suite is shown in Table 8. As can be seen in this table, CoDE used the least CPU time. However, the solution quality of the algorithm is relatively poor, as can be seen in Table 7. For other algorithms, the computational time

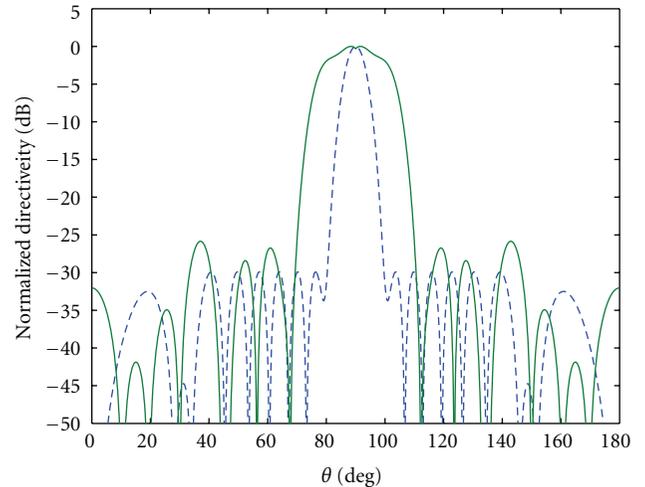


FIGURE 8: Dual-beam array pattern (Experiment II) with coupling effect.

is in the same quantity level, but DE/BBO can produce best solutions among all algorithms.

8. Conclusions

Application of hybrid differential evolution with biogeography-based optimization for the reconfigurable antenna array with quantized phase shifter is discussed in this paper. The effectiveness of the proposed algorithm is demonstrated on the design of a reconfigurable array antenna without and with the quantized phase excitations. The effect of the quantization in the continuous formulation of phased excitation is presented. In order to reduce the effect of mutual coupling between the antenna-array elements, the dynamic range ratio is minimized. The experimental results clearly

indicate superior performance of the proposed algorithm in comparison to some recent optimization algorithms. We hope that this paper sparks a new venue of research in the problem of solving reconfigurable antenna array.

Acknowledgments

This research is fully supported by the National Natural Science Foundation of China under Grant no. 60803102 and also funded by NSFC Major Research Program 60496321: Basic Theory and Core Techniques of Non-Canonical Knowledge.

References

- [1] O. M. Bucci, G. D'Elia, G. Mazzarella, and G. Panariello, "Antenna pattern synthesis. A new general approach," *Proceedings of the IEEE*, vol. 82, no. 3, pp. 358–371, 1994.
- [2] O. M. Bucci, G. Mazzarella, and G. Panariello, "Reconfigurable arrays by phase-only control," *IEEE Transactions on Antennas and Propagation*, vol. 39, no. 7, pp. 919–925, 1991.
- [3] M. Dürr, A. Trastoy, and F. Ares, "Multiple-pattern linear antenna arrays with single prefixed amplitude distributions: modified Woodward-Lawson synthesis," *Electronics Letters*, vol. 36, no. 16, pp. 1345–1346, 2000.
- [4] Y. Rahmat-Samii and E. Michielssen, *Electromagnetic Optimization by Genetic Algorithms*, John Wiley & Sons, New York, NY, USA, 1999.
- [5] A. Alphones and V. Passoupathi, "Null steering in phased arrays by positional perturbations: a genetic algorithm approach," in *Proceedings of the IEEE International Symposium on Phased Array Systems and Technology*, pp. 203–207, October 1996.
- [6] S. Baskar, A. Alphones, and P. N. Suganthan, "Design of reconfigurable antenna array using improved multi-agent genetic algorithms," in *Asia-Pacific Microwave Photonics Conference (APMP '04)*, New Delhi, India, 2004.
- [7] D. Gies and Y. Rahmat-Samii, "Particle swarm optimization for reconfigurable phase-differentiated array design," *Microwave and Optical Technology Letters*, vol. 38, no. 3, pp. 168–175, 2003.
- [8] D. W. Boeringer and D. H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 3, pp. 771–779, 2004.
- [9] K. Güneş and A. Akdağlı, "Null steering of linear antenna arrays using a modified tabu search algorithm," *Progress In Electromagnetics Research*, vol. 30, pp. 167–182, 2001.
- [10] S. Baskar, A. Alphones, and P. N. Suganthan, "Genetic-algorithm-based design of a reconfigurable antenna array with discrete phase shifters," *Microwave and Optical Technology Letters*, vol. 45, no. 6, pp. 461–465, 2005.
- [11] F. Dobias and J. Gunther, "Reconfigurable array antennas with phase-only control of quantized phase shifters," in *Proceedings of the 45th IEEE Vehicular Technology Conference*, pp. 35–39, Chicago, Ill, USA, July 1995.
- [12] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.
- [13] R. S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.
- [14] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Computing*, vol. 9, no. 6, pp. 448–462, 2005.
- [15] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
- [16] D. H. Kim, A. Abraham, and J. H. Cho, "A hybrid genetic algorithm and bacterial foraging approach for global optimization," *Information Sciences*, vol. 177, no. 18, pp. 3918–3937, 2007.
- [17] D. H. Choi, "Cooperative mutation based evolutionary programming for continuous function optimization," *Operations Research Letters*, vol. 30, no. 3, pp. 195–201, 2002.
- [18] C. Zhang, J. Ning, S. Lu, D. Ouyang, and T. Ding, "A novel hybrid differential evolution and particle swarm optimization algorithm for unconstrained optimization," *Operations Research Letters*, vol. 37, no. 2, pp. 117–122, 2009.
- [19] X. H. Shi, Y. C. Liang, H. P. Lee, C. Lu, and L. M. Wang, "An improved GA and a novel PSO-GA-based hybrid algorithm," *Information Processing Letters*, vol. 93, no. 5, pp. 255–261, 2005.
- [20] J. Sun, Q. Zhang, and E. P. K. Tsang, "DE/EDA: a new evolutionary algorithm for global optimization," *Information Sciences*, vol. 169, no. 3–4, pp. 249–262, 2005.
- [21] N. Noman and H. Iba, "Accelerating differential evolution using an adaptive local search," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 107–125, 2008.
- [22] W. Gong, Z. Cai, and L. Jiang, "Enhancing the performance of differential evolution using orthogonal design method," *Applied Mathematics and Computation*, vol. 206, no. 1, pp. 56–69, 2008.
- [23] M. G. H. Omran, A. P. Engelbrecht, and A. Salman, "Differential evolution based particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '07)*, vol. 4, pp. 112–119, April 2007.
- [24] F. Neri and V. Tirronen, "Scale factor local search in differential evolution," *Memetic Computing*, vol. 1, no. 2, pp. 153–171, 2009.
- [25] Z. Yang, J. He, and X. Yao, "Making a difference to differential evolution," in *Advances in Metaheuristics for Hard Optimization*, Z. Michalewicz and P. Siarry, Eds., pp. 397–414, Springer, Berlin, Germany, 2008.
- [26] W. Gong, Z. Cai, and C. X. Ling, "DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization," *Soft Computing*, vol. 15, no. 4, pp. 645–665, 2010.
- [27] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous space," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [28] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, 2008.
- [29] J. Teo, "Exploring dynamic self-adaptive populations in differential evolution," *Soft Computing*, vol. 10, no. 8, pp. 673–686, 2006.
- [30] Y. Wang, Z. Cai, and Q. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 55–66, 2011.
- [31] J. Zhang and A. C. Sanderson, "JADE: adaptive differential evolution with optional external archive," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

