Progress In Electromagnetics Research B, Vol. 21, 87–111, 2010

OPTIMAL SYNTHESIS OF LINEAR ANTENNA ARRAYS WITH MULTI-OBJECTIVE DIFFERENTIAL EVOLUTION

S. Pal

Department of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata 700032, India

B.-Y. Qu

School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore 639798, Singapore

S. Das

Department of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata 700032, India

P. N. Suganthan

School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore 639798, Singapore

Abstract—Linear antenna array design is one of the most important electromagnetic optimization problems of current interest. In contrast to a plethora of recently published articles that formulate the design as the optimization of a single cost function formed by combining distinct and often conflicting design-objectives into a weighted sum, in this work, we take a Multi-objective Optimization (MO) approach to solve the same problem. We consider two design objectives: the minimum average Side Lobe Level (SLL) and null control in specific directions that are to be minimized simultaneously in order to achieve the optimal spacing between the array elements. Our design method employs a recently developed and very competitive multi-objective evolutionary

Corresponding author: S. Das (swagatamdas19@yahoo.co.in).

algorithm called MOEA/D-DE that uses a decomposition approach for converting the problem of approximation of the Pareto Fronts (PF) into a number of single objective optimization problems. This algorithm employs Differential Evolution (DE), one of the most powerful real parameter optimizers in current use, as the search method. As will be evident from the shape of the approximated PFs obtained with MOEA/D-DE, the two design-objectives are in conflict and usually, performance cannot be improved significantly for one without deteriorating the other. Unlike the single-objective approaches, the MO approach provides greater flexibility in the design by yielding a set of equivalent final solutions from which the user can choose one that attains a suitable trade-off margin as per requirements. We illustrate that the best compromise solution attained by MOEA/D-DE can comfortably outperform state-of-the-art single-objective algorithms like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Tabu Search Algorithm (TSA), and Memetic Algorithm (MA). In addition, we compared the results obtained by MOEA/D-DE with those obtained by one of the most widely used MO algorithm called NSGA-2 and another generic multi-objective DE variant that uses non-dominated sorting, on the basis of the R-indicator, hypervolume indicator, and quality of the best trade-off solutions obtained.

1. INTRODUCTION

Antenna arrays play an important role in detecting and processing signals arriving from different directions. Compared with a single antenna that is limited in directivity and bandwidth, an array of sensors can have its beam-pattern modified with an amplitude and phase distribution called the weights of the array. After preprocessing the antenna outputs, signals are weighted and summed to give the antenna array beam-pattern. The antenna array pattern synthesis problem consists of finding weights that satisfy a set of specifications on the beam-pattern [1–3].

The goal in antenna array geometry synthesis is to determine the physical layout of the array, which can produce a radiation pattern that is closest to the desired pattern. The shape of the desired pattern can vary widely depending on the application. Many synthesis methods are concerned with suppressing the Side Lobe Level (SLL) while preserving the gain of the main beam [4]. Other methods deal with the null control to reduce the effects of interference and jamming. For the linear array geometry, this can be done by designing the spacing between the elements, while keeping a uniform excitation over the array aperture. Other methods of controlling the array pattern use non-uniform excitation and phased arrays [1].

It is well known that the classical derivative-based optimization techniques need a starting point that is reasonably close to the final solution, or they are likely to be stuck in a local minimum. As the number of parameters and hence the size of the solution space increases, the quality of the solution strongly depends on the estimation of the initial values. If the initial values fall in a region of the solution space where all the local solutions are poor, a local search is limited to finding the best of these poor solutions. The computational drawbacks of existing numerical methods have forced the researchers all over the world to rely on metaheuristic algorithms founded on simulations of some natural phenomena to solve antenna problems. These algorithms use an objective function, optimization of which leads to the side lobe suppression and null control [5]. Metaheuristic algorithms such as GAs [6–8], Simulated Annealing (SA) [9], Tabu Search Algorithm (TSA) [10], Taguchi's method [11], Memetic Algorithms (MAs) [12], and PSO [13, 14] have already used in the design of antenna arrays, e.g., see [5, 15–19].

Most of the existing metaheuristic design approaches combine (like the works reported in [5, 15–19]) separate objectives (which are often conflicting) through a weighted linear sum into a single aggregated objective function. The weighted sum method is however, subjective and the solution obtained will depend on the values (more precisely, the relative values) of the weights specified. It is hard, if not impossible, to find a universal set of weights, that click on different instantiations of the same problem. Motivated by the inherent multi-objective nature of the antenna array design problems and the overwhelming growth in the field of Multi-Objective Evolutionary Algorithms (MOEAs), we started to look for the most recently developed MOEAs that could solve the linear array synthesis problem much more efficiently as compared to the conventional single-objective approaches. Since Differential Evolution (DE) [20–22] has emerged as one of the most powerful stochastic real-parameter optimizers of current interest and unlike PSOs and GAs, has not been used extensively in antenna design context, we were also looking for the state-of-the-art MO variants of DE, when our search converged to a decomposition-based MOEA, called MOEA/D-DE [23, 24], that ranked first among 13 state-of-the-art MOEAs in the unconstrained MOEA competition held under the IEEE Congress on Evolutionary Computation (CEC) 2009 [25]. MOEA/D-DE uses DE as its main search strategy and decomposes an MO problem into a number of scalar optimization sub-problems to optimize them simultaneously. Each sub-problem is optimized by only using information from its several neighboring sub-problems and this feature considerably reduces the computational complexity of the algorithm.

In this work, we employ MOEA/D-DE to optimize spacing between the elements of a linear array while achieving the best possible trade-off between the two desgin objectives corresponding to minimum average Side Lobe Level (SLL) and null placement. The best trade-off solutions, identified with a fuzzy membership function based approach [26] over three difficult instances of the design problem are shown to outperform the solutions achieved with other published metaheuristic approaches using PSO, MA, GA, and TSA. The results of MOEA/D-DE are also compared to those obtained with a generic MO variant of DE (MODE) that uses non-dominated sorting and one of the most widely known MO algorithm called NSGA-2 [27], which was applied to the linear array design by Panduro et al. in [28]. However, from what follows, it will be evident that the approach of [28] is completely different from that taken in this paper.

The rest of the paper is organized in the following way. In Section 2, we briefly outline the basic DE, MOEA/D-DE, and MODE along with the constraint handling technique. Formulation of the linear array pattern synthesis as a multi-objective optimization problem has been discussed in Section 3. Section 4 describes the experimental settings, presents the experimental results and discusses their implications. Section 5 finally concludes the paper and unfolds a few important future research issues.

2. MULTI-OBJECTIVE OPTIMIZATION, MODE, AND MOEA/D-DE — BRIEF BACKGROUNDS

2.1. General MO Problems

Due to the multiple criteria nature of most real-world problems, MO problems are ubiquitous, particularly throughout engineering applications [29–32]. As the name indicates, this class of problems involves multiple objectives, which should be optimized simultaneously and that are very often in conflict with each other. Two key concepts related to the solutions of MO problems are *dominance* and *Pareto-optimality* and below we present them in a formal way [33]:

Definition 1: Consider without loss of generality the following MO problem with D decision variables x (parameters) and M objectives y:

$$\text{Minimize } \vec{Y} = f\left(\vec{X}\right) = \left(f_1(x_1, \ldots, x_D), \ldots, f_M(x_1, \ldots, x_D)\right), \quad (1)$$

where $\vec{X} = [x_1, \ldots, x_D]^T \in P$ and $\vec{Y} = [y_1, \ldots, y_M]^T \in O$ and $\vec{X}$ is called decision (parameter) vector, P is the parameter space, $\vec{Y}$ is the objective vector, and O is the objective space. A decision vector $\vec{A} \in P$ is said to dominate another decision vector $\vec{B} \in P$ (also written as $\vec{A} \prec \vec{B}$ for minimization) if and only if:

$$\forall i \in \{1, \ldots, M\} : \ f_i\left(\vec{A}\right) \leq f_i\left(\vec{B}\right) \wedge \exists j \in \{1, \ldots, M\} : \ f_j(\vec{A}) < f_j(\vec{B}) \tag{2}$$

Based on this convention, we can define non-dominated, *Pareto-optimal* solutions as follows:

Definition 2: Let $\vec{A} \in P$ be an arbitrary decision vector.

(a) The decision vector $\vec{A}$ *is said to be* non-dominated regarding the set $P' \subseteq P$ if and only if there is no vector in P' which can dominate $\vec{A}$.

(b) The decision (parameter) vector $\vec{A}$ is called Pareto-optimal if and only if $\vec{A}$ is non-dominated regarding the whole parameter space P.

Under constrained MO problems, Equation (1) is minimized subjected to the following constraints:

1) Inequality Constraints:

$$g_r\left(\vec{X}\right) = g_l(x_1, x_2, \ldots, x_D) \geq 0, \quad r = 1, 2, \ldots, R; \tag{3}$$

2) Equality Constraints:

$$h_k\left(\vec{X}\right) = h_k(x_1, x_2, \ldots, x_D) = 0, \quad k = 1, 2, \ldots, K; \tag{4}$$

3) Boundary Constraints:

$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \ldots, D. \tag{5}$$

While dealing with constrained MO problems, individuals that satisfy all of the constraints are called feasible individuals while individuals that do not satisfy at least one of the constraints are called infeasible individuals.

2.2. DE and MODE

DE is a population-based optimization algorithm that was first introduced by Price and Storn [20] in 1995 for solving single-objective optimization problems. It has been extended to solve MO problems [34–38], frequently with an external archive to hold the current set of non-dominated solutions. The overall structure of Multi-Objective Differential Evolution (MODE) is similar to other MOEAs. MODE starts with a randomly initialized population of size NP that is evolved by applying the DE operators such as mutation, recombination and selection to every population member until a stopping criterion is satisfied. For each parent $\vec{X}_p$ at each generation, an associated mutant vector $\vec{V}_p$ can be generated by using one of the five mutation strategies [22] — DE/rand/1, DE/best/1, DE/current-to-best/2, DE/best/2, DE/rand/2. The strategy used here in MODE and also in MOEA/D-DE (to be outlined next) is DE/rand/1:

$$\vec{V}_p = \vec{X}_{r1} + F \cdot \left(\vec{X}_{r2} - \vec{X}_{r3}\right) \tag{6}$$

where r_1, r_2, and r_3 are random and mutually different integers generated in the range [1, NP (population size)], which should also be different from the current trial vector. F is a scale factor in $[0, 2]$ used to scale the differential vectors. The "binomial" (i.e., uniform) crossover operation is applied to each pair of the generated mutant vector and its corresponding trial vector. The process can be represented for the i-th component of each vector as:

$$u_{p,i} = \begin{cases} v_{p,i}, & \text{if } (rand_i \leq CR \text{ or } j = j_{rand}) \\ x_{p,i}, & \text{otherwise,} \end{cases} \tag{7}$$

where u_p is the trial vector. Crossover rate CR determines which dimensions of $\vec{U}_p$ are copied from $\vec{V}_p$ or $\vec{X}_p$ respectively, where $rand_i$ is uniformly distributed in the range [0, 1]. A new population is formed by combining the current solutions and newly generated solutions. Since we are using binomial crossover, schemes (6) and (7) are jointly known as DE/rand/1/bin (bin for binomial). This new population will go through a selection process to choose parents for the next generation. The selection process is different for single objective and multi-objective cases, as in single objective case the comparison between parent and offspring is based on a single objective value, and the vector yielding the smaller result is kept for the next generation (minimization problems). In the presence of multiple objectives, the selection process may be accomplished through non-domination sorting [33]. After the process of non-domination sorting, the population members will be ranked according to the front number. The top solutions will be selected to fill the external archive. The top solutions in the external archive are chosen as the parents in the next generation.

2.3. The MOEA/D-DE Algorithm

A multi-objective evolutionary algorithm based on decomposition was first introduced by Zhang and Li in 2007 [39] and extended with DE-based reproduction operators in [23, 24]. Instead of using non-domination sorting for different objectives, the MOEA/D algorithm decomposes a multi-objective optimization problem into a number of single objective optimization sub-problems by using weight vectors λ and optimizes them simultaneously. Each sub-problem is optimized by sharing information between its neighboring sub-problems with similar weight values. MOEA/D uses Tchebycheff decomposition approach [40] to convert the problem of approximating the Pareto front into a number of scalar optimization problems. Let $\vec{\lambda}^1, \ldots, \vec{\lambda}^N$ be a set of evenly spread weight vectors and $\vec{Y}^* = (y_1^*, y_2^*, \ldots, y_M^*)$ be a reference point, i.e., for minimization problem, $y_i^* = \min\{f_i(\vec{X})|\vec{X} \in \Omega\}$ for each $i = 1, 2, \ldots, M$. Then the problem of approximation of the PF can be decomposed into N scalar optimization sub-problems by Tchebycheff approach and the objective function of the j-th sub-problem is:

$$g^{te}\left(\vec{X}|\vec{\lambda}^j, \vec{Y}^*\right) = \max_{1 \le i \le M}\left\{\lambda_i^j \left|f_i(x) - y_i^*\right|\right\}, \tag{8}$$

where $\vec{\lambda}^j = \left(\lambda_1^j, \ldots, \lambda_M^j\right)^T$, $j = 1, \ldots, N$ is a weight vector, i.e., $\lambda_i^j \ge 0$ for all $i = 1, 2, \ldots, m$ and $\sum_{i=1}^{m} \lambda_i^j = 1$.

MOEA/D minimizes all these N objective functions iteratively in a single run. Neighborhood relations among these single objective sub-problems are defined based on the distances among their weight vectors. Each sub-problem is then optimized by using information mainly from its neighboring sub-problems. In MOEA/D, the concept of neighborhood, based on similarity between weight vectors with respect to Euclidean distances, is used to update the solution. The neighborhood of the i-th sub-problem consists of all the sub-problems with the weight vectors from the neighborhood of $\vec{\lambda}^i$.

At each generation, the MOEA/D maintains following variables:

1. A population ($\vec{X}^1, \ldots, \vec{X}^N$) with size N, where $\vec{X}_i$ is the current solution to the i-th sub-problem.
2. The fitness values of each population corresponding to a specific sub-problem.
3. The reference point $\vec{Y}^* = (y_1^*, y_2^*, \ldots, y_M^*)$, where y_i^* is the best value found so far for objective i.
4. An external population (EP), which is used to store non-dominated solutions found during the search.

The MOEA/D-DE algorithm is schematically presented in Table 1.

Table 1. The MOEA/D-DE algorithm.

1. Initialization	Initialize the External Population (EP)
	Compute the Euclidean distances between any two weight vectors and find out the T closest weight vectors to each weight vector where T is the neighborhood size.
	Randomly generate an initial population $\vec{X}^1, \ldots, \vec{X}^N$ and evaluate the fitness values.
	Initialize the reference points by a problem-specific method.
2. Update	Reproduction: reproduce the offspring $\vec{U}$ by DE/rand/1/bin scheme.
	Repair: Repair the solution if $\vec{U}$ is out of the boundary and the value is reset to be a randomly selected value inside the boundary.
	Update of reference points, if the fitness value of $\vec{U}$ is better than the reference point.
	Update the neighboring solutions, if the fitnessvalue of $\vec{U}$ is better.
	Update of EP by removing all the vectors that are dominated by $\vec{U}$ and add $\vec{U}$ to EP if no vector in EP dominates it.
3. Stopping Criteria	If stopping criteria is satisfied, then stop and output EP. Otherwise, go to Step 2

2.4. Constraint Handling

In order to avoid mutual coupling and the appearance of grating lobes the decision variables are needed to obey certain constraints in the MO linear array design problem detailed in the next section. Here we use a constraint handling technique based on *Superiority of the Feasible solution* (SF). This kind of constraint handling method was

Table 2. Fitness modification by using the SF method.

Step 1:	Extract feasible solution (s) from the combined population. If there is no feasible solution, the new fitness value is overall constraint violation value. If all solutions are feasible, the fitness values are unchanged.
Step 2:	For $m = 1$ to M *(number_of_objectives)* Find the maximum objective values for each objective among the feasible solutions. End For
Step 3:	Form the new fitness values for infeasible solutions using Equation (9) and keep the fitness values of feasible solutions unchanged.

first introduced by Powell and Skolnick [41] and then extended by Deb [42] for single-objective optimization. The SF as used in multi-objective optimization can be expressed in the following way:

$$fitness_m\left(\vec{X}\right) = \begin{cases} f_m\left(\vec{X}\right) & \text{if } \vec{X} \text{ is feasible,} \\ f_{warst}^m + v\left(\vec{X}\right) & \text{otherwise,} \end{cases} \tag{9}$$

where f_{warst}^m is the m-th objective value of the worst feasible solution with respect to objective m in the population and $v(\vec{X})$ is the overall constraint violation. If there is no feasible solution in the current population, f_{warst}^m is zero. In this approach, there is no penalty factor involved. Feasible solutions are always better than the infeasible ones. In this way, the infeasible solutions are expected to evolve towards the feasible region and feasible solutions are expected to evolve towards the global optimal front. Table 2 shows how the fitness values are modified by the superiority of feasible solution method.

3. MULTI-OBJECTIVE FORMULATION OF THE DESIGN PROBLEM

An antenna array is a configuration of individual radiating elements that are arranged in space and can be used to produce a directional radiation pattern. For a linear antenna array, let us assume that we have $2N$ isotropic radiators placed symmetrically along the z-axis. The array geometry of a 6-element linear array is shown in Figure 1.

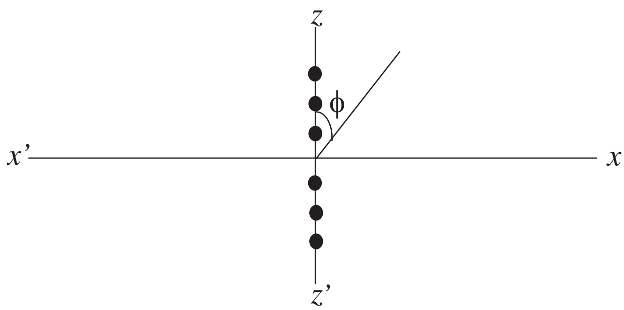

Figure 1. Symmetrically placed 6-element linear array.

The radiation pattern of the array depends on the geometric configuration, the distance between the elements, the amplitude and phase excitation of the elements and also the radiation pattern of individual antenna elements [43]. Since the array elements are identical, we can assume the radiation pattern of the array considering the sum of all contributions signals of each individual element. The above relation is often referred to as *pattern multiplication*, which indicates that the total field of the array is equal to the product of the field due to the single element located at the origin and a factor called array factor, AF, defined as [1, 2]:

$$AF(\varphi) = 2 \cdot \sum_{n=1}^{N} I_n \cdot \cos\left[k \cdot z_n \cdot \cos(\varphi) + \phi_n\right], \tag{10}$$

where $k = \frac{2\pi}{\lambda}$ = Wave number, I_n = Excitation magnitude of the nth element, φ_n = Phase of the nth element, and z_n = Location of the nth element along z-axis.

If we further assume a uniform excitation of amplitude and phase (that is $I_n = 1$ and $\varphi_n = 0$ for all elements), the array factor can be further simplified as:

$$AF(\varphi) = 2 \cdot \sum_{n=1}^{N} \cos\left[k \cdot z_n \cdot \cos(\varphi)\right] \tag{11}$$

Now the statement of the problem, addressed here, simply reduces to: apply an evolutionary algorithm to find the locations z_n of the array elements that will result in an array beam with minimum SLL and nulls at specific directions. There are, however, certain constraints regarding element spacing. Adjacent elements if spaced too closely can lead to mutual coupling effects and if spaced too far off can lead to appearance of grating lobe [1, 28]. Thus the element spacing needs to be constrained. In this problem the element spacing z_n is normalized with respect to $\lambda/2$. The constraints that need to be considered for normalized element spacing z'_n is given below.

$$0.5 \leq z'_{n+1} - z'_n \leq 1, \quad n \in [1, N-1] \tag{12}$$

The first element along positive z-axis needs to be placed such that it is neither too close nor too far from the first element on the negative z-axis. Thus the constraint for the first element is given below in (13):

$$0.3 \leq z_1 \leq 0.5 \tag{13}$$

For side lobe suppression, the objective function is:

$$f_1 = \sum_i \frac{1}{\Delta\phi_i} \int_{\phi_{li}}^{\phi_{ui}} |AF(\phi)|^2 \cdot d\phi \tag{14}$$

And for null control we propose to use:

$$f_2 = \sum_k |AF(\phi_k)|^2 \tag{15}$$

For a given beamwidth finding the right balance between side lobe suppression and null control has always been a issue. However when we are using an MO algorithm then we get an approximation of the PF, which contain many solutions. Either a designer can choose a desired solution from the approximated PF or he can follow a standard procedure for finding the best compromise solution. We have used a fuzzy membership function based technique [26] for the above purpose. In either case an MOEA allows us greater flexibility in designing a linear antenna of our choice because a single-objective EA gives us only one solution in one run which might not completely cater to the designer's needs.

4. EXPERIMENTAL SETUP AND RESULTS

This Section compares the performance of three MO algorithms: MOEA/D-DE, MODE, and NSGA-2, and four state-of-the-art single-objective algorithms Genetic Algorithm (GA), Tabu Search Algorithm (TSA), Memetic Algorithm (MA), and Particle Swarm Optimization (PSO) over three instantiations of the linear array design problem.

4.1. Performance Indices for MO Algorithms

For comparison among the MO algorithms, we used the following two performance indices apart from the shape of the approximated PF obtained from each of the algorithm.

(1) R **indicator** (I_{R2}) **[44]**: It can be expressed as

$$I_{R2} = \frac{\sum_{\lambda \in \Lambda} u^* (\lambda, A) - u^* (\lambda, R)}{|\Lambda|}, \tag{16}$$

where R is a reference set, u^* is the maximum value reached by the utility function u with weight vector λ on an approximation set A, i.e., $u^* = \max_{y \in A} u_\lambda (y)$. We choose the augmented Tchebycheff function as the utility function.

(2) **Hypervolume difference to a reference set** ($I_{\bar{H}}$) **[44]**: The hypervolume indicator I_H measures the hypervolume of the objective space that is weakly dominated by an approximation set A, and is to be maximized. Here we consider the hypervolume difference to a reference set R, and we will refer to this indicator as $I_{\bar{H}}$, which is defined as $I_{\bar{H}} = I_H(R) - I_H(A)$ where smaller values correspond to higher quality as opposed to the original hypervolume I_H.

4.2. Determining the Best Compromise Solution

An MO algorithm usually returns a series of non-dominated solutions, unlike a unique solution as returned by a single-objective algorithm. For comparison with the published single-objective optimization approaches, we used a fuzzy membership function based technique to determine the best compromise solution from the approximated PFs obtained with MOEA/D-DE, MODE, and NSGA-2 in the following way. While selecting the best trade-off solution from a set of non-dominated solutions, a Decision Maker's (DM) judgment may very often turn out to be imprecise and under such circumstances it is natural to assume that the DM may have fuzzy or imprecise goals for each objective functions. The fuzzy sets are characterized by membership functions having values from 0 to 1 and indicating the degree of belongingness of an entity to a certain fuzzy set. In the present method, the DM defines a membership function μ_i for each objective f_i by taking account of the minimum and maximum values of each objective function together with the rate of increase of membership satisfaction. Following [26, 45], we assume that μ_i is a strictly monotonic decreasing and continuous function defined as:

$$\left.\mu_i \begin{array}{ll} = 1 & \text{if } f_i \le f_i^{\min} \\ = \frac{f_i^{\max} - f_i}{f_i^{\max} - f_i^{\min}} & \text{if } f_i^{\min} < f_i < f_i^{\max} \\ = 0 & \text{if } f_i \ge f_i^{\max} \end{array} \right\} \tag{17}$$

where $f_i^{\min}$ and $f_i^{\max}$ are the minimum and maximum value of the i-th objective solution among all non-dominated solutions, respectively.

The value of the membership function indicates how much (in the scale from 0 to 1) a non-inferior solution has satisfied the i-th objective. The sum of the membership function values (μ_i, $i = 1, \ldots, M$) for all the objectives can be calculated for measuring the 'accomplishment' of each solution in satisfying the objectives. The 'accomplishment' of each non-dominated solution can be rated with respect to all the N_s non-dominated solutions by normalizing its 'accomplishment' over the sum of the 'accomplishments' of the N_s non-dominated solutions in the following way [46]:

$$\mu^q = \frac{\sum_{i=1}^{N_{obj}} \mu_i^q}{\sum_{k=1}^{N_s} \sum_{i=1}^{N_{obj}} \mu_i^k}. \tag{18}$$

The function μ^q in (18) can be treated as a membership function for non-dominated solutions, in a fuzzy set and represented as a fuzzy cardinal priority ranking of the non-dominated solutions. The best compromise solution is then the one having the maximum value of μ^q [46].

4.3. Parametric Setup

For MODE, as control parameters we took $F = 0.35$, $CR = 0.2$, and $NP = 200$. For the NSGA-2 algorithm, we took crossover probability $= 0.9$, mutation probability $= 1/N$ ($2N$ being the total number of antenna elements in the array),distribution index for crossover $= 20$, distribution index for mutation $= 20$. In MOEA/D-DE, for the DE operator we took $F = 0.5$, $CR = 1$, distribution index $\eta = 20$, and the mutation rate $p_m = {}^1/_D$ as per [23]. For the competitor algorithms PSO, GA, MA, and TSA, we kept the best possible parametric setup as explained in the relevant literatures [5, 16]. We used the

Table 3. Geometry of the 22 element linear array normalized with respect to $\lambda/2$ (design example 1).

ALGORITHM	ELEMENT SPACING										
MOEA/D-DE	±0.327	±0.827	±1.305	±1.984	±2.541	±3.045	±3.902	±4.405	±5.369	±6.205	±7.384
NSGA - 2	±0.3000	±0.8015	±1.302	±1.844	±2.526	±3.067	±4.013	±5.012	±6.023	±7.025	±8.212
MODE	±0.3000	±0.800	±1.300	±1.827	±2.532	±3.055	±3.991	±4.895	±5.899	±6.912	±8.095
GA	±0.4000	±0.941	±1.523	±2.031	±2.534	±3.083	±3.719	±4.340	±5.164	±5.737	±6.708
TSA	±0.4000	±0.902	±1.477	±1.996	±2.497	±3.123	±3.716	±4.317	±5.167	±6.040	±6.796
PSO	±0.3006	±1.177	±1.855	±2.685	±3.524	±4.428	±5.468	±6.580	±7.953	±9.552	±11.00
MA	±0.4000	±0.938	±1.462	±2.006	±2.529	±3.067	±3.762	±4.323	±5.202	±6.057	±7.048

same parametric setup for an algorithm over all the instantiations of the design problem. Source codes for NSGA-II were obtained from http://www.iitk.ac.in/kangal/codes.shtml and codes for MOEA/D-DE were taken from http://cswww.essex.ac.uk/staff/qzhang/.

Table 4. Best, worst, mean, and standard deviations of the performance metrics for comparing the MO algorithms over design example 1.

Performance Metric	Value-type	MOEA/D-DE	MODE	NSGA-2
R-indicator	Best	**8.31e-09**	1.27e-05	1.28e-05
	Worst	**3.62e-05**	0.0206	0.116
	Mean	**1.04e-05**	0.005	0.0247
	Std. Dev.	**1.48e-05**	0.0083	0.0409
Hypervolume -indicator	Best	**3.09e-11**	1.42e-05	1.75e-05
	Worst	**8.87e-05**	0.0592	0.3566
	Mean	**2.71e-05**	0.0143	0.0745
	Std. Dev.	**3.34e-05**	0.0237	0.1246

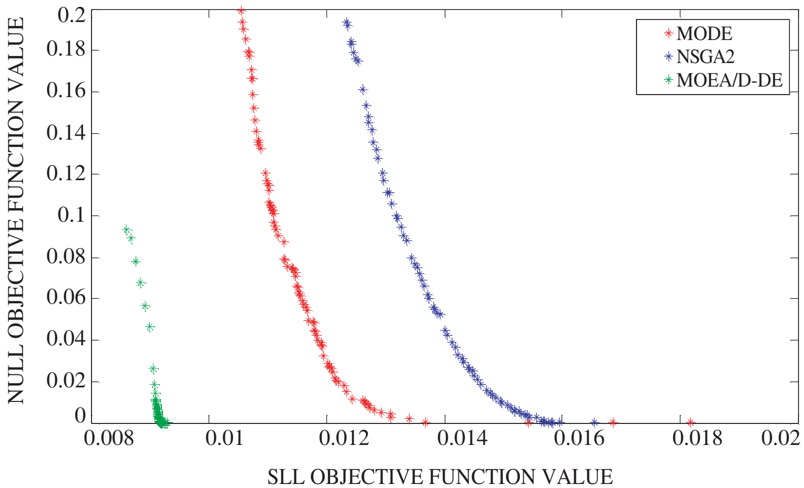

Figure 2. Final approximations of the Pareto fronts produced by MOEA/D-DE, MODE, and NSGA-2 (design example 1).

Table 5. Design objectives and directivity achieved with six algorithms (design example 1).

ALGORITHM	SLL (f_1)	NULL (f_2)	$f_1 + f_2$	DIRECTIVITY IN dB
MOEA/D-DE	**0.00920**	2.596e-05	**0.00922**	**17.3123**
MODE	0.0136	**1.802e-05**	0.0136	17.2321
NSGA-2	0.0157	7.004e-04	0.0164	16.2131
GA	0.0231	0.0020	0.0251	16.1113
PSO	0.0127	0.0056	0.0183	16.8425
TSA	0.0294	0.00012	0.0295	15.1011
MA	0.0199	5.3089e-05	0.0200	16.6474

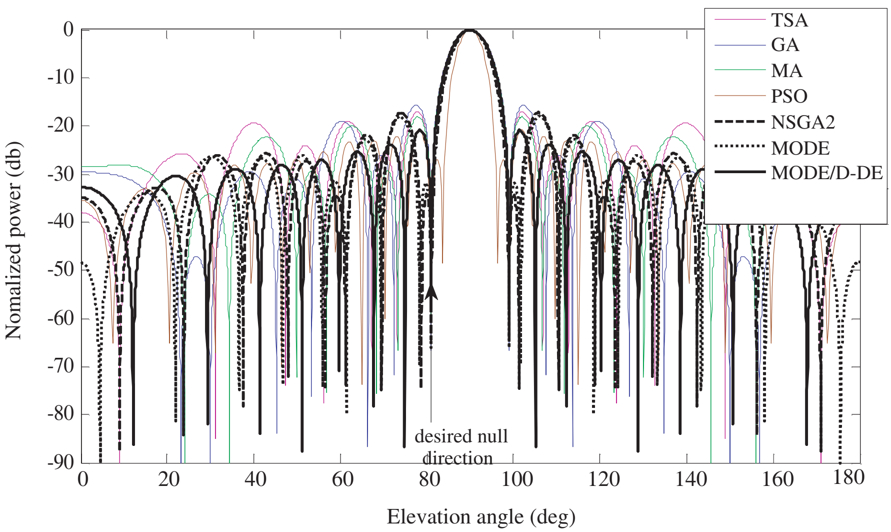

Figure 3. Array patterns obtained for design example 1.

4.4. Results

Three instantiations of the design problem were considered and the results for each are provided below. In what follows we provide the best results obtained over 25 independent trials of each algorithm. Each run of each algorithm was continued up to 3×10^5. Function Evaluations (FEs) in order to make the comparison fair enough.

Design Example 1: This example amounts to the design of a 22 element array, which has minimum SLL in bands $[0°, 82°]$ and

Table 6. Geometry of the 26 element linear array normalized with respect to $\lambda/2$ (design example 2).

ALGORTIHM	ELEMENT SPACING												
MOEA/D-DE	±0.346	±0.854	±1.395	±2.121	±2.627	±3.243	±4.094	±4.544	±5.533	±6.172	±7.144	±8.149	±9.291
MODE	±0.300	±1.037	±1.689	±2.336	±3.185	±3.686	±4.671	±5.225	±6.195	±6.981	±7.978	±8.989	±10.176
NSGA-2	±0.305	±1.199	±1.829	±2.561	±3.421	±3.969	±5.005	±5.555	±6.580	±7.307	±8.307	±9.285	±10.407
TSA	±0.486	±1.431	±2.416	±3.187	±4.113	±4.822	±5.725	±6.548	±7.503	±8.004	±8.907	±9.411	±10.404
PSO	±0.400	±1.347	±2.035	±3.011	±3.836	±4.765	±5.699	±6.555	±7.457	±8.247	±9.241	±10.20	±11.20
GA	±0.500	±1.406	±2.334	±3.266	±4.034	±4.897	±5.568	±6.443	±7.244	±8.195	±9.132	±10.06	±10.99
MA	±0.400	±1.102	±1.925	±2.666	±3.637	±4.635	±5.620	±6.563	±7.517	±8.432	±9.404	±10.35	±11.30

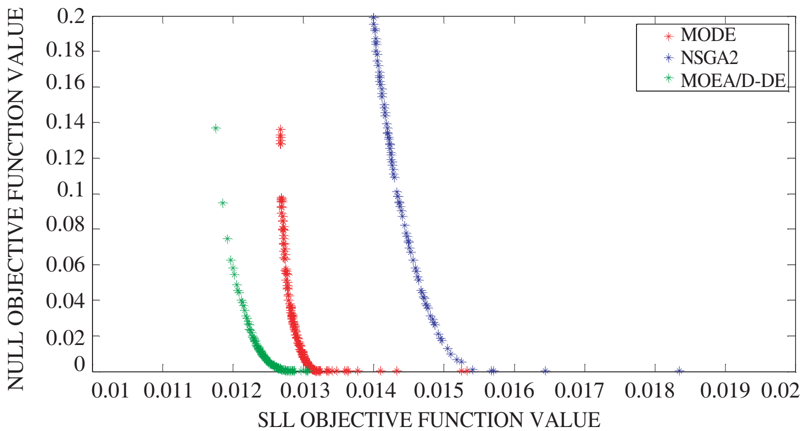

Figure 4. Final approximations of the Pareto fronts produced by MOEA/D-DE, MODE, and NSGA-2 (design example 2).

$[98°, 180°]$ and null direction in 81°. Table 3 reports the array geometry normalized to $\lambda/2$. Table 4 reports the values of the R-indicator and the hypervolume indicator for the three MO algorithms over design example 1. The approximated PFs obtained with MOEA/D-DE, MODE, and NSGA2 have been shown in Figure 2. Table 5 provides the final values of the two objective functions f_1 and f_2 (corresponding to SLL and Null placement respectively) as obtained with the three MO algorithms (best compromise solution for each) as well as four population-based single-objective optimization algorithms, the combined cost function $f_1 + f_2$ that acts as objective function for PSO, GA, TA, and MA algorithms, and directivities (in dB) obtained with the seven competitor algorithms. The best entries in Tables 4 and 5 have been marked in bold. Finally Figure 3 shows the

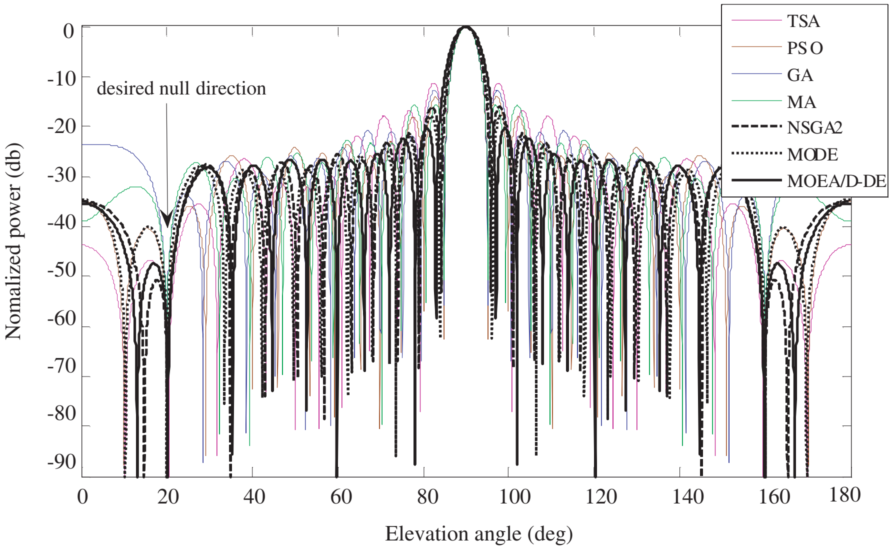

Figure 5. Array patterns obtained for design example 2.

Table 7. Best, worst, mean, and standard deviations of the performance metrics for comparing the MO algorithms over design example 2.

Performance Metric	Value-type	MOEA/D-DE	MODE	NSGA-2
R-indicator	Best	**7.45e-07**	1.18e-05	2.65e-05
	Worst	**5.44e-05**	7.60e-05	1.23e-04
	Mean	**2.28e-05**	3.78e-05	8.07e-05
	Std. Dev.	**1.75e-05**	2.07e-05	2.85e-05
Hypervolume -indicator	Best	1.78e-05	**6.11e-07**	7.85e-05
	Worst	**1.71e-04**	2.27e-04	3.04e-04
	Mean	**8.20e-05**	1.14e-04	2.07e-04
	Std. Dev.	**4.70e-05**	7.46e-05	7.27e-05

normalized power (in dB) versus elevation angle (in degrees) plot for seven algorithms over the design example 1.

The shape of the approximated PFs obtained with the MO algorithms as shown in Figure 2 indicates that the two design objectives

Table 8. Design objectives and directivity achieved with seven algorithms (design example 2).

ALGORITHM	SLL (f_1)	NULL (f_2)	$f_1 + f_2$	DIRECTIVITY IN dB
MOEA/D-DE	**0.01278**	**1.341e-05**	**0.01278**	**17.812**
MODE	0.0132	1.446e-05	0.0132	17.6116
NSGA-2	0.0156	8.541e-05	0.0333	16.1218
GA	0.0293	0.00051	0.0294	15.9193
PSO	0.0196	9.8684e-005	0.0196	16.641
TSA	0.0330	0.00067	0.0336	15.3191
MA	0.0244	0.00023	0.0246	16.0142

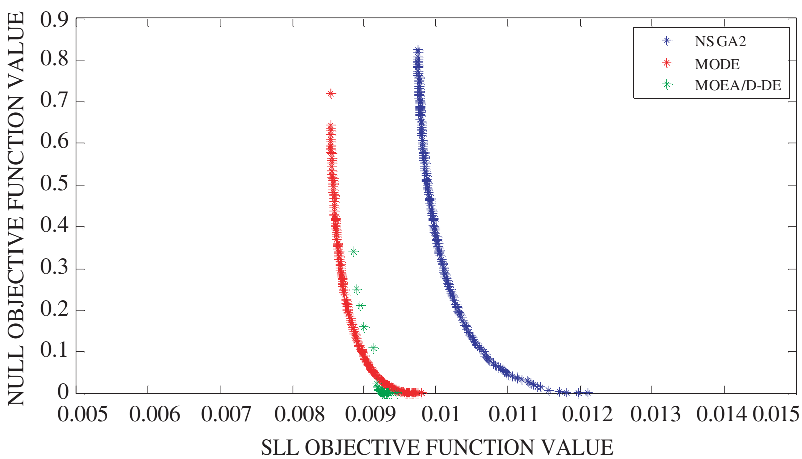

Figure 6. Final approximations of the Pareto fronts produced by MOEA/D-DE, MODE, and NSGA-2 (design example 3).

are conflicting in nature, a feature that makes the problem more suitable for the application of MO techniques. Table 4 indicates that MOEA/D-DE achieves best performance among the MO algorithms. A close inspection of Table 5 reveals that the principal lobe requirement is best achieved by MOEA/D-DE. The null control obtained with MODE is the best and slightly better than MOEA/D-DE. Although the single-objective algorithms used $f_1 + f_2$ as their cost function, the value of $f_1 + f_2$ is best obtained with MOEA/D-DE that also yields the best value of directivity for the linear array synthesized.

Design example 2: In this case we are required to design a 26 element array which has minimum SLL in bands $[0°, 82°]$ and $[98°, 180°]$ and null direction in $20°$. The best results obtained with seven contestant algorithms have been reported in Tables 6, 7, and 8 that correspond to Tables 3, 4, and 5 of design example 1 and Figures 4 and 5 that are similar in theme to Figures 2 and 3 of the previous design example.

Figure 4 indicates that MOEA/D-DE can achieve better trade-off solutions (with lower values of both the objectives) than NSGA-2 and MODE. Table 6 shows that indeed MOEA/D-DE could outperform all other algorithms compared in terms of both the objectives as well as the directivity. From the plot of array patterns shown in Figure 5, it is obvious that MOEA/D-DE suppresses the side lobes to the greatest extent. Moreover, it also gives lowest gain at the desired null of $20°$.

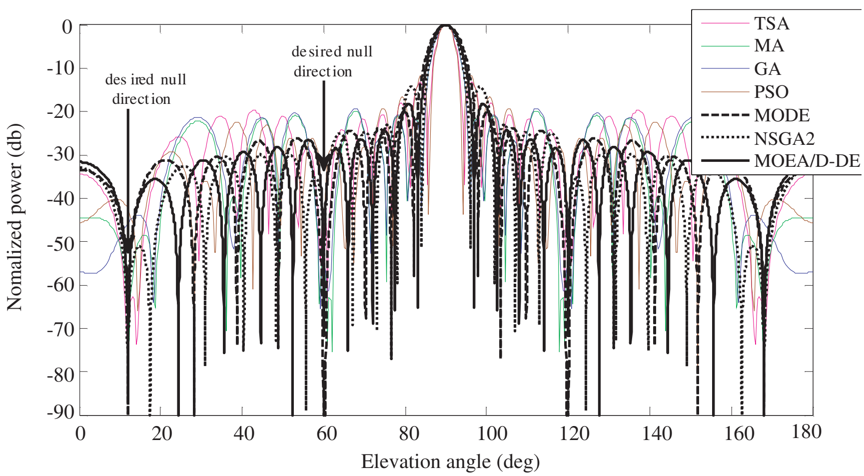

Figure 7. Array patterns obtained for design example 3.

Table 9. Geometry of the 26 element linear array normalized with respect to $\lambda/2$ (design example 3).

ALGORITHM	ELEMENT SPACING											
MOEA/D-DE	±0.3863	±0.886	±1.520	±2.212	±2.714	±3.406	±4.152	±4.653	±5.597	±6.134	±7.123	±8.049	±9.249
MODE	±0.4152	±1.0931	±1.869	±2.719	±3.296	±4.086	±4.771	±5.358	±6.331	±6.892	±7.892	±8.652	±9.851
NSGA-2	±0.3021	±0.816	±1.383	±1.885	±2.388	±2.905	±3.587	±4.132	±5.023	±5.521	±6.409	±7.230	±8.427
GA	±0.4242	±0.8472	±1.579	±2.468	±2.993	±4.391	±4.629	±5.640	±6.399	±7.791	±8.795	±9.974	±11.38
PSO	±0.5798	±1.741	±2.806	±3.923	±4.885	±5.939	±7.100	±8.137	±9.171	±9.956	±10.75	±11.82	±13.00
TSA	±0.5311	±1.491	±2.468	±3.445	±4.524	±5.501	±6.491	±7.268	±8.498	±9.514	±10.27	±11.44	±12.42
MA	±0.4521	±0.8512	±1.606	±2.497	±3.019	±4.397	±4.629	±5.687	±6.399	±7.792	±8.796	±9.976	±11.40

Design example 3: In this case we are required to design a 26 element array which has minimum SLL in bands $[0°, 80°]$ and $[100°, 180°]$ and null direction in $12°, 60°$. The best results obtained with seven contestant algorithms have been reported in Tables 9, 10, and 11 that correspond to Tables 6, 7, and 8 of design example 2 and Figures 6 and 7 that are similar in theme to Figures 4 and 5 of the previous design example.

Table 10. Best, worst, mean, and standard deviations of the performance metrics for comparing the MO algorithms over design example 3.

Performance Metric	Value-type	MOEA/D-DE	MODE	NSGA-2
R-indicator	Best	**4.43e-06**	6.98e-06	1.09e-05
	Worst	**3.38e-05**	3.30e-05	9.27e-05
	Mean	**1.40e-05**	2.06e-05	3.45e-05
	Std. Dev.	**1.15e-05**	7.56e-06	2.79e-05
Hypervolume -indicator	Best	**1.12e-06**	1.85e-06	3.55e-05
	Worst	**3.88e-05**	1.16e-04	1.86e-04
	Mean	**1.97e-05**	7.81e-05	8.47e-05
	Std. Dev.	**1.49e-05**	3.32e-05	6.43e-05

Table 11. Design objectives and directivity achieved with six algorithms (design example 3).

ALGORITHM	SLL (f_1)	NULL (f_2)	$f_1 + f_2$	DIRECTIVITY IN dB
MOEA/D-DE	**0.00937**	**6.99e-06**	**0.00938**	**17.9313**
MODE	0.00971	3.79e-05	0.00974	17.9121
NSGA-2	0.0119	1.105e-04	0.0120	17.6740
GA	0.0300	0.0133	0.0433	17.0084
PSO	0.0198	0.1153	0.1351	17.8855
TSA	0.0300	0.0463	0.0763	17.8319
MA	0.0253	0.0011	0.0264	17.0162

Tables 10 and 11 indicate that in this case also MOEA/D-DE yields superior results as compared to all other competitor algorithms.

5. CONCLUSION

In this work we model the linear antenna array synthesis as an MO problem where the design objectives are the Side Lobe Level (SLL) and null control in specific directions. One of the recently developed and state-of-the-art MO variant of DE, called MOEA/D-DE, has been used to solve three significant instantiations of the design problem and the results have been compared against those obtained with a Pareto-dominance based MODE and another very prominent MO algorithm called NSGA-2 as well as four single-objective optimization techniques namely PSO, GA, TA, and MA. MOEA/D-DE is shown to yield better trade-off curves between the two design-objectives for linear antenna arrays with non-uniform separations as compared to MODE and NSGA-2. Also MOEA/D-DE returned better values of the R-indicator and hypervolume indicator over all the design instances as compared to its other two MO competitors. Over the three reported cases, MOEA/D-DE and MODE yielded better values of the combined cost function, which was also minimized by the single-objective optimization algorithms. The array obtained with MOEA/D-DE also produced maximum directivity as compared to the five competitor algorithms over all the problems. In contrast to single-objective approaches, the MO algorithms finally provides a set of design solutions that could allow the practitioner to satisfy the gain requirements of wireless communications systems without having to increase the number of elements in the array, thereby reducing the array cost and the control complexity.

Future research may focus on achieving more control of the array pattern by using MOEA/D-DE to optimize, not only the location, but also the excitation amplitude and phase of each element in the array. The MO approach may also be extended in future to other more complex array geometries. Design of monopulse antenna arrays, has so far been addressed only with single objective techniques that minimize a linear combination of different design objectives. We believe that the MO formulation will improve the synthesis of the difference patterns of monopulse antenna over the existing techniques.

REFERENCES

1. Godara, L. C., *Handbook of Antennas in Wireless Communications*, CRC, Boca Raton, FL, 2002.

2. Bucci, O. M., D. D'Elia, G. Mazzarella, and G. Panatiello, "Antenna pattern synthesis: A new general approach," *Proc. IEEE*, Vol. 82, 358–371, 1994.
3. Rahmat-Samii, Y. and E. Michielssen, *Electromagnetic Optimization by Genetic Algorithms*, Wiley, New York, 1999.
4. Lebret, H. and S. Boyd, "Antenna array pattern synthesis via convex optimization," *IEEE Transactions on Signal Processing*, Vol. 45, No. 3, 526–532, March 1997.
5. Khodier, M. M. and C. G. Christodoulou, "Linear array geometry synthesis with minimum side lobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, Vol. 53, No. 8, 2674–2679, August 2005.
6. Holland, J. H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Harbor, 1975.
7. Bäck, T., D. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*, Oxford Univ. Press, 1997.
8. Eiben, A. E. and J. E. Smith, *Introduction to Evolutionary Computing*, Springer, 2003.
9. Kirkpatrik, S., C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, Vol. 220, 671–680, 1983.
10. Glover, F. and M. Laguna, *Tabu Search*, Kluwer, Norwell, MA, 1997.
11. Taguchi, G., S. Chowdhury, and Y. Wu, *Taguchi's Quality Engineering Handbook*, Wiley, New York, 2005.
12. Ong, Y.-S. and A. J. Keane, "Meta-lamarckian learning in memetic algorithms," *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 2, 99–110, 2004.
13. Kennedy, J. and R. Eberhart, "Particle swarm optimization," *Proc. IEEE Int. Conf. Neural Networks,* 1942–1948, 1995.
14. Kennedy, J., R. C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann, San Francisco, CA, 2001.
15. Udina, A., N. M. Martin, and L. C. Jain, "Linear antenna array optimization by genetic means," *Third International Conference on Knowledge-based Intelligent Information Engineering Systems Adelaide*, Australia, September 1999.
16. Cengiz, Y. and H. Tokat, "Linear antenna array design with use of genetic, memetic and tabu search optimization algorithms," *Progress In Electromagnetics Research C*, Vol. 1, 63–72, 2008.
17. Weng, W.-C., F. Yang, and A. Z. Elsherbeni, "Linear antenna array synthesis using taguchi's method: A novel optimization technique in electromagnetics," *IEEE Transactions on Antennas*

and Propagation, Vol. 55, No. 3, 723–730, March 2007.

18. Ares-Pena, F. J., A. Rodriguez-Gonzalez, E. Villanueva-Lopez, and S. R. Rengarajan, "Genetic algorithms in the design and optimization of antenna array patterns," *IEEE Transactions on Antennas and Propagation*, Vol. 47, 506–510, March 1999.
19. Tian, Y. B. and J. Qian, "Improve the performance of a linear array by changing the spaces among array elements in terms of genetic algorithm," *IEEE Transactions on Antennas and Propagation*, Vol. 53, 2226–2230, July 2005.
20. Storn, R. and K. V. Price, "Differential evolution — A simple and efficient adaptive scheme for global optimization over continuous spaces," Technical Report TR-95-012, ICSI, http://http.icsi.berkeley.edu/∼storn/litera.html, 1995.
21. Storn, R. and K. Price, "Differential evolution — A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, Vol. 11, No. 4, 341–359, 1997.
22. Price, K., R. Storn, and J. Lampinen, *Differential Evolution — A Practical Approach to Global Optimization*, Springer, Berlin, 2005.
23. Li, H. and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE Trans. on Evolutionary Computation*, Vol. 12, No. 2, 284–302, 2009.
24. Zhang, Q., W. Liu, and H. Li, "The performance of a New MOEA/D on CEC09 MOP test instances," *Proceedings of the Eleventh Conference on Congress on Evolutionary Computation*, 203–208, Trondheim, Norway, May 18–21, 2009.
25. Zhang, Q., A. Zhou, S. Z. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Technical Report CES-887, University of Essex and Nanyang Technological University, 2008.
26. Abido, M. A., "A novel multiobjective evolutionary algorithm for environmental/economic power dispatch," *Electric Power Systems Research*, Vol. 65, 71–81, Elsevier, 2003.
27. Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, 182–197, 2002.
28. Panduroa, M. A., D. H. Covarrubiasa, C. A. Brizuelaa, and F. R. Maranteb, "A multi-objective approach in the linear antenna array design," *Int. J. Electron. Commun., (AEÜ)*, Vol. 59, 205–212, 2005.
29. Reklaitis, G. V., A. Ravindran, and K. M. Ragsdell, *Engineering*

Optimization: Methods and Applications, 2nd Edition, John Wiley, New Jersey, 2006.
30. Abraham, A., L. C. Jain, and R. Goldberg, *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, Springer Verlag, London, 2005.
31. Carlos, A. C. C. and G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms*, World Scientific, 2004.
32. Nedjah, N. and L. D. M. Mourelle, *Real-world Multi-objective System Engineering*, Illustrated Edition, Nova Science Publishers, Deccermber 2005.
33. Deb, K., *Multi-objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, 2001.
34. Xue, F., A. C. Sanderson, and R. J. Graves, "Pareto-based multi-objective differential evolution," *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, Vol. 2, 862–869, IEEE Press, Canberra, Australia, 2003.
35. Robic, T. and B. Filipic, "DEMO: Differential evolution for multiobjective optimization," *Evolutionary Multi-Criterion Optimization, Third International Conference, EMO 2005*, C. A. Coello Coello, A. H. Aguirre, and E. Zitzler, Vol. 3410, 520–533, Springer, Guanajuato, Mexico, 2005.
36. Iorio, A. W. and X. Li, "Solving rotated multi-objective optimization problems using differential evolution," *AI 2004: Advances in Artificial Intelligence, Proceedings*, Vol. 3339, 861–872, Springer-Verlag, 2004.
37. Huang, V. L., A, K. Qin, P. N. Suganthan, and M. F. Tasgetiren, "Multi-objective optimization based on self-adaptive differential evolution algorithm," *Congress on Evolutionary Computation (CEC'2007)*, 3601–3608, Singapore, September 2007.
38. Zamuda, A., J. Brest, B. Boskovic, and V. Zumer, "Differential evolution for multiobjective optimization with self adaptation," *Congress on Evolutionary Computation (CEC'2007)*, 3617–3624, Singapore, September 2007.
39. Zhang, Q. and H. Li, "MOEA/D: A multi-objective evolutionary algorithm based on decomposition," *IEEE Trans. on Evolutionary Computation*, Vol. 11, No. 6, 712–731, 2007.
40. Miettinen, K., *Nonlinear Multiobjective Optimization*, Kuluwer Academic Publishers, 1999.
41. Powell, D. and M. M. Skolnick, "Using genetic algorithms in engineering design optimization with non-linear constraints," *Proceedings of the Fifth International Conference on Genetic*

Algorithms, San Mateo, CA, USA, 1993.

42. Deb, K., "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, 311–338, 2000.
43. Van Veen, B. D. and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE Acoust. Speech Signal Process. Mag.*, Vol. 5, 4–24, 1988.
44. Knowles, J., L. Thiele, and E. Zitzler, "A tutorial on the performance assessment of stochastic multiobjective optimizers," *Computer Engineering and Networks Laboratory*, ETH Zurich, Switzerland, Feb. 2006.
45. Dhillon, J. S., S. C. Parti, and D. P. Kothari, "Stochastic economic emission load dispatch," *Electric Power Syst. Res.*, 179–186, 1993.
46. Tapia, C. G. and B. A. Murtagh, "Interactive fuzzy programming with preference criteria in multiobjective decision making," *Comput. Oper. Res.*, Vol. 18, 307–316, 1991.