MOBES: A Multiobjective Evolution Strategy for Constrained Optimization Problems

To Thanh Binh & Ulrich Korn

Institute of Automation, University of Magdeburg, Germany

Abstract. In this paper a new MultiOBjective Evolution Strategy (MOBES) for solving multi-objective optimization problems subject to linear and nonlinear constraints is presented. MOBES is based on the new concept of $\mathcal{C}$-, $\mathcal{F}$- and $\mathcal{N}$-fitness, which allows systematically to handle constraints and (in)feasible individuals. The existence of niche infeasible individuals in every population enables to explore new areas of the feasible region and new feasible pareto-optimal solutions. Moreover, MOBES proposed a new selection algorithm for searching, maintaining a set of feasible pareto-optimal solutions in every generation. The performance of the MOBES can be successfully evaluated on two selected test problems.

1 Introduction

1.1 Multiobjective optimization problem

The general multiobjective optimization problem with linear and nonlinear constraints can be formally stated as below:

$$\min_{\mathbf{x}} \mathbf{f}(\mathbf{x}) = \min_{\mathbf{x}} (f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_{\mathrm{N}}(\mathbf{x}))$$

where $\mathbf{x} = (x_1, x_2, \cdots, x_{\mathrm{n}})^T \in \mathcal{F} \subseteq \mathcal{S} \subseteq \mathcal{R}^{\mathrm{n}}$.
N and n are the number of objective functions and the number of variables, respectively.
The *search space* $\mathcal{S}$ is a region of the n-dimensional space $\mathcal{R}^n$, for example, a n-dimensional rectangle defined by lower and upper bounds of variables:

$$x_{\mathrm{i}}^{(\mathrm{lower})} \leq x_{\mathrm{i}} \leq x_{\mathrm{i}}^{(\mathrm{upper})}, \ \forall i = \overline{1, n}$$

whereas the *feasible region* $\mathcal{F} \subseteq \mathcal{S}$ is defined by a set of m additional linear and nonlinear constraints ($m \geq 0$):

$$g_{\mathrm{j}}(\mathbf{x}) \leq 0, \ \forall j = \overline{1, q}$$
$$h_{\mathrm{j}}(\mathbf{x}) = 0, \ \forall j = \overline{q+1, m}$$

Without loss of generality, in this paper only multiobjective optimization problems with m constraints in terms of inequations will be taken into account, because by computational implementation a constraint in terms of equation like $h_{\mathrm{j}}(\mathbf{x}) = 0$ can be replaced by a pair of inequalities:

$$\begin{cases} -h_{\mathrm{j}}(\mathbf{x}) - \epsilon & \leq 0, \\ h_{\mathrm{j}}(\mathbf{x}) - \epsilon & \leq 0, \end{cases}$$

where an additional parameter ϵ is used to define the precision of the system. Therefore it seems to be a special case of constraints in terms of inequalities.

To solve this problem is to find all feasible trade-offs among the multiple, conflicting objectives, known as a set $\mathcal{P}_{\mathrm{x}}$ of *feasible pareto-optimal solutions* ($\mathcal{P}_{\mathrm{x}} \subseteq \mathcal{F}$) in the variable space [4]:

Definition 1 (Pareto-Optimality). A vector $\mathbf{x}^* \in \mathcal{F}$ is said to be feasible pareto-optimal if and only if there exists no other vector $\mathbf{x} \in \mathcal{F}$ such that:

$$\begin{cases} f_{\mathrm{i}}(\mathbf{x}) \leq f_{\mathrm{i}}(\mathbf{x}^*) & \forall i = \overline{1, N} \\ f_{\mathrm{j}}(\mathbf{x}) < f_{\mathrm{j}}(\mathbf{x}^*) & \text{for at least one } j \end{cases} \quad (1)$$

Definition 2 (Inferiority). The vector $\mathbf{u} = (f_1(\mathbf{x}^*), f_2(\mathbf{x}^*), \cdots, f_N(\mathbf{x}^*))$ satisfying (1) is said to be inferior to the vector $\mathbf{v} = (f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_N(\mathbf{x}))$.

Definition 3 (Non-Inferiority). Vectors $\mathbf{u}$ and $\mathbf{v}$ are said to be non-inferior to each other if neither $\mathbf{u}$ is inferior to $\mathbf{v}$ nor $\mathbf{v}$ is inferior to $\mathbf{u}$.

$\mathcal{P}_x$ corresponds to a set $\mathcal{P}_f$ of nondominated or noninferior vectors (solutions) lying on a surface known as a *pareto-optimal surface* in the objective function space.

Because $\mathcal{P}_f$ often contains an infinite number of elements, the solving multiobjective optimization problems namely leads to find and maintain a representative sampling of solutions on the pareto-optimal surface.

1.2 Literature survey

Recently some evolutionary algorithms (EAs) for solving multiobjective optimization problems are proposed [10, 5, 3, 8, 6, 1, 2]. They have not produced a meaningfull breakthrough in the area of constrained multiobjective optimization problems due to the fact that they have not addressed the handling of constraints and objectives in a systematic way. Some of them consider multiobjective and constrained optimization separetely, both in general terms and in the context of evolutionary algorithms [3]. Therefore they try to convert the above constrained multiobjective optimization either into a multiobjective optimization without constraints (e.g. using penalty function methods [8]) or into constrained optimization. The others are based on evolution strategies and provide a well approximation of pareto-optimal solutions [6, 1, 2]. The weakness of these methods is that it is necessary to initialize a population by feasible individuals. The finding feasible individuals is itself a difficult problem especially in cases the ratio between the feasible and search region is small. MOBES is proposed to overcome this disadvantage of the traditional multiobjective ESs and to enable simultaneously to handle both constraints and multiple objectives.

This paper presents a systematic way to handle constraints as well as (in)feasible individuals in the next two sections. In section 4 a global selection scheme allowing to maintain a representative sampling of solutions on the pareto-optimal surface is briefly discussed. Finally, two test cases are shown to illustrate a remarkable efficiency of the MOBES.

2 Handling constraints

Similar to traditional ESs, each individual consists of a vector of objective variables $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ (a point in the search space), a strategy parameter vector $\mathbf{s} = (s_1, s_2, \cdots, s_n)^T$ (a vector of standard deviations). To evaluate the fitness of an individual the two following measures have to be taken into account:

- an objective function vector $\mathbf{f}(\mathbf{x})$ (so-called $\mathcal{F}$-fitness in the objective function space),
- a degree of violation of constraints (it is called $\mathcal{C}$-fitness in the constraint space).

The $\mathcal{F}$-fitness of an individual can be described by a point (vector) in the N-dimensional objective function space. The remaining problem is how to evaluate the $\mathcal{C}$-fitness of an individual.

Let $c_i(\mathbf{x}) = \max\{g_i(\mathbf{x}), 0\}$, $\forall i = \overline{1, m}$, the $\mathcal{C}$-fitness of an individual can be charaterized by a vector:

$$\mathcal{C}(\mathbf{x}) = (c_1(\mathbf{x}), c_2(\mathbf{x}), \cdots, c_m(\mathbf{x})), \tag{2}$$

(a point in the constraint space $\mathcal{R}_c^m$). Clearly, $\mathcal{C} \equiv \mathbf{0}$ for feasible individuals and $\mathcal{C} > \mathbf{0}$ for infeasible ones (at least one element of $\mathcal{C}$ is bigger than zero). Using this measure the original point of the constraint space corresponds to the feasible region of the search space. This description of the $\mathcal{C}$-fitness allows precisely to represent an individual in the constraint space and to determine its degree of infeasibility. It was used for the old version of the MOBES. The difficulties with this description are as below:

- When the number of constraints is large, a large amount of memory is necessary to save a vector $\mathcal{C}(\mathbf{x})$ for every individual. This problem also increasingly arises for handling many multiple objectives because the $\mathcal{F}$-fitness is itself a vector.
- For two infeasible individuals with objective variable vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ so that $\mathcal{C}(\mathbf{x}_1)$ and $\mathcal{C}(\mathbf{x}_2)$ are noninferior it is very difficult to know which individual is better than the other. In other words, the ranking infeasible individuals should be performed by using noninferiority (comparision between infeasible individuals essentially is comparision between vectors $\mathcal{C}(\mathbf{x})$). For this reason, it takes much time for ranking the whole infeasible population.

To avoid them instead of using a vector $(c_1(\mathbf{x}), c_2(\mathbf{x}), \cdots, c_{\mathrm{m}}(\mathbf{x}))$ a scalar value determining the *distance* between a point $(c_1(\mathbf{x}), c_2(\mathbf{x}), \cdots, c_{\mathrm{m}}(\mathbf{x}))$ and the original point in the constraint space is used for evaluating a degree of (in)feasibility [9]:

$$\mathcal{C}(\mathbf{x}) = \Big(\sum_{i=1}^{m} [c_{\mathrm{i}}(\mathbf{x})]^p \Big)^{\frac{1}{p}}, \quad (p > 0). \tag{3}$$

The second measure of the $\mathcal{C}$-fitness has experimentally shown to be as good as the first one and acceptable for multiobjective constrained optimization problems. Therefore it is used to design the MOBES. Using this concept, the population can be divided into classes (so-called $\mathcal{C}$-classes) corresponding to degrees of (in)feasibility. It is clear, the 0-class includes all feasible individuals; individuals of the higher classes are "*farther*" from the feasible region than ones of the lower classes.

3 Handling (in)feasible individuals

The MOBES allows mutation and reproduction operators to generate both feasible and infeasible offspring. Therefore it is necessary to check whether offspring are better than their parent (by mutation and reproduction) or to select better individuals for the next generation (by selection). For highly constrained problems, a population can still consist of all infeasible individuals through first several generations (no feasible solution can be found), the first priority is given to making an infeasible population to evolve towards the feasible region. To do it the $\mathcal{C}$-fitness is used to rank infeasible individuals; "better infeasible" individuals belong to lower classes (i.e. they are "*nearer*" to the feasible region).

Criterion 4 (Selection Between Classes). An individual of the $\mathcal{C}_1$-class ($\mathcal{C}_1 > 0$) is said to be better than the other of the $\mathcal{C}_2$-class if and only if $\mathcal{C}_1 < \mathcal{C}_2$.

To avoid problems for which an infeasible population converges to (possible) local minima of the $\mathcal{C}$-fitness, like traditional ESs some so-called niche infeasible individuals type 1 should be selected and inserted into a new population:

Criterion 5 (Niche Infeasible Individuals Type 1). If there is no feasible individual in a population, niche infeasible individuals type 1 should have an as small value of the $\mathcal{C}$-fitness as possible and be as far as possible from the "best current infeasible" individual.

Mathematically, a niche infeasible individual type 1 should have a small value of the so-called niche fitness (denoted by $\mathcal{N}_{\mathrm{i}}$-fitness):

$$\mathcal{N}_{\mathrm{i}}(\mathbf{x}) = \frac{\mathcal{C}(\mathbf{x}) - \mathcal{C}(\mathbf{x}_{ibest})}{||\mathbf{x} - \mathbf{x}_{ibest}||^{\beta}}, \quad (\mathbf{x} \neq \mathbf{x}_{ibest})$$

where:

- $\mathbf{x}_{ibest}$ and $\mathcal{C}(\mathbf{x}_{ibest})$ are an objective variable vector and the $\mathcal{C}$-fitness of the "best infeasible" individual, respectively.
- $||.||$ denotes a norm in the n-dimensional parameter space.
- β is a scalar value ($\beta = 1, 2, \cdots$).

Notice that traditional ESs [1] have also shown that the use of niche individuals may help the ES in finding many global minima with the same objective function value located at different points of the variable space. For constrained optimization problems for which the feasible region is disconnected the $\mathcal{C}$-fitness (see *Eq. 3*) has the (global) smallest value 0 at different subsets of the feasible region. Therefore it is meaningful to use niche infeasible individuals to explore other subsets of the feasible region.

Individuals of the same class can be compared together using their $\mathcal{F}$-fitness and the concept of pareto-optimality. So a current pareto-optimal surface can be developed in every generation:

Criterion 6 (Selection In A Class). Among individuals of the same class, better individuals are non-inferior ones (that means, their $\mathcal{F}$-fitness vectors are noninferior to each other).

During the search process for the feasible region the population at some stage of the evolution process may contain some feasible and infeasible individuals. It is necessary (but not easy) to introduce a criterion for comparing between feasible and infeasible individuals. For many optimization problems the feasible region is non-convex or the ratio between the feasible and search region is too small so that no feasible offspring can be generated even from feasible individuals through many generations. But it is hopeful that infeasible individuals lying near to the feasible pareto-optimal surface can generate feasible offspring which are even better than offspring of some feasible individuals (for example, for a non-convex feasible region, arithmetical crossover operators [7] for one infeasible and one feasible individual can generate feasible offspring whereas no feasible individual can be found by using these operators for both feasible parents). That means, any feasible individual *is not always better* than any infeasible one. It is reasonable that infeasible individuals co-exist in a population with other feasible ones and infeasible offspring lying in a neigbourhood of the feasible region may be better than other feasible individuals. The difficulty is that which infeasible individuals are said to be "better" than feasible ones and how to choose them from a population. This problem seems to be a problem happened before making a decision for using niche individuals in traditional ESs [1, 6]. Therefore, niche infeasible individuals seem to be "better" than feasible ones.

Criterion 7 (Niche Infeasible Individuals Type 2). If at least one feasible individual exists in the population, niche infeasible individuals type 2 should have an as small value of the $\mathcal{C}$-fitness as possible and be as far as possible from the centroid of a feasible subpopulation (a set of feasible individuals, denoted by $\mathcal{J}$).

The niche fitness of infeasible individuals (denoted by $\mathcal{N}_{if}$-fitness) can be evaluated by:

$$\mathcal{N}_{if}(\mathbf{x}) = \frac{\mathcal{C}(\mathbf{x})}{||\mathbf{x} - \mathbf{z}||^{\beta}}, \tag{4}$$

where $\mathbf{z} = \frac{\sum_{\mathbf{x} \in \mathcal{J}} \mathbf{x}}{nF}$ is the centroid of a set $\mathcal{J}$ (nF - the number of individuals of $\mathcal{J}$).

Our initial experiments showed that this criterion was useful for many optimization problems.

Instead of directly comparing between feasible and infeasible individuals the MOBES would like to introduce the following criteria to generate infeasible individuals and then to select niche infeasible individuals in a population:

Criterion 8 (Extra class). Infeasible individuals up to the $\mathcal{C}_{\text{extra}}$-class (i. e. individuals of $\mathcal{C}$-classes so that $0 < \mathcal{C} \leq \mathcal{C}_{\text{extra}}$) are said to be in the same class (called the *extra class*).

Criterion 9 (Extension Class). An infeasible offspring of a feasible individual is said to be viable if and only if it belongs to the so-called extension $\mathcal{C}_{\text{extension}}$-class defined by:

$$\mathcal{C}_{\text{extension}} = \max\{\mathcal{C}_{\text{extra}}, \mathcal{C}_{\text{pop}}\},$$

where $\mathcal{C}_{\text{pop}}$ is the highest class in the population (corresponding to the infeasible individual with the biggest distance to the original point of the constraint space).

Criterion 10 (Selection of Infeasible Individuals). Every population must contain at least a given number (nI) of (niche) infeasible individuals.

The problem of how to choose the best value of $\mathcal{C}_{\text{extra}}$ and nI is far from trivial. With a too small value of $\mathcal{C}_{\text{extra}}$ it is difficult to generate offspring in the extra class. Otherwise the choosing a higher value of $\mathcal{C}_{\text{extra}}$ leads to generate more infeasible individuals (individuals of higher classes). If nI is too big or too small, the population would slowly converge to the set of feasible pareto-optimal solutions. Our first experiments showed that the value $\mathcal{C}_{\text{extra}}$=0.1 and $nI = 5\% nP$ were preferable, where nP is the population size.

4 Maintaining a feasible pareto-optimal set

In the previous section, many operators were used to find the feasible region and to shift then the population towards the feasible pareto-optimal surface. After some generations there exist more feasible individuals on the feasible pareto-optimal surface (noninferior individuals) than necessary. A problem how to maintain a representative sampling of solutions (to create an uniform distribution of the population) on the pareto-optimal surface has therefore to be considered.

An algorithm for solving this problem was proposed in the old versions of the MOBES [2]. The weakness of this algorithm is that the density of the population in a neigbourhood of the selfish minima (an own minimum of each objective function) is still higher than in other regions of the pareto-optimal surface. For this reason it should be slighly modified as belows:

Criterion 11. Let

$$\begin{aligned}
\mathbf{f}^{(\text{min})} &= (\min f_1, \min f_2, \cdots, \min f_{\text{N}}) \\
&= (f_1^{(\text{min})}, f_2^{(\text{min})}, \cdots, f_{\text{N}}^{(\text{min})}) \\
\mathbf{f}^{(\text{max})} &= (\max f_1, \max f_2, \cdots, \max f_{\text{N}}) \\
&= (f_1^{(\text{max})}, f_2^{(\text{max})}, \cdots, f_{\text{N}}^{(\text{max})}),
\end{aligned}$$

where the minimum and maximum operators are performed along each coordinate axes of the objective function space for all feasible noninferior individuals of the population. Then, the current feasible trade-offs surface is bound in a hyperparallelogram defined by vectors $\mathbf{f}^{(\text{min})}$ and $\mathbf{f}^{(\text{max})}$.

- Dividing each interval $[f_{\text{i}}^{(\text{min})}, f_{\text{i}}^{(\text{max})}]$ into N_{pop} (the desired number of feasible individuals per population) small sections (denoted by $\mathcal{H}_{\text{j}}^{\text{i}}$, $j = \overline{1, N_{\text{pop}}}$) with the length δ_{i}, i. e.:

  $$\delta_{\text{i}} = \frac{f_{\text{i}}^{(\text{max})} - f_{\text{i}}^{(\text{min})}}{N_{\text{pop}}}.$$

  Individuals in the section $\mathcal{H}_{\text{j}}^{\text{i}}$ have a lower value of the *i-th* objective function than one of individuals of the section $\mathcal{H}_{\text{k}}^{\text{i}}$, $\forall k > j$.
- Evaluating the density of the population per section (the number of noninferior individuals on a section).
- Along the *i-th* coordinate axes of the objective function space ($i = \overline{1, N}$) the best individual in each of the $\dfrac{N_{\text{pop}}}{N + k}$ first sections is selected, where k is an integer number.
- Other individuals can be selected from remaining sections with the lowest density.

5 Test Cases

For all two test cases the following important parameters of the MOBES were used:

- Population size = 100
- Number of niche infeasible individuals = 5
- Number of parents for mutation and reproduction = 10
- Number of offspring per mutation = 5.

5.1 Test Case 1

The problem [8] is minimizing $(f_1(\mathbf{x}), f_2(\mathbf{x}))$:

$$f_1(\mathbf{x}) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$$
$$f_2(\mathbf{x}) = 9x_1 - (x_2 - 1)^2$$

subject to non-linear constraints:

$$x_1^2 + x_2^2 - 225 \leq 0$$
$$x_1 - 3x_2 + 10 \leq 0$$

and bounds: $-20 \leq x_i \leq 20, \ \forall i = 1, 2$.
A set of pareto-optimal solutions is found after running 5 generations. In opposition to [8] the current population more quickly runs to the pareto-optimal frontier (29 generations in [8]) and is more uniformly distributed on it (see *Fig. 1*).

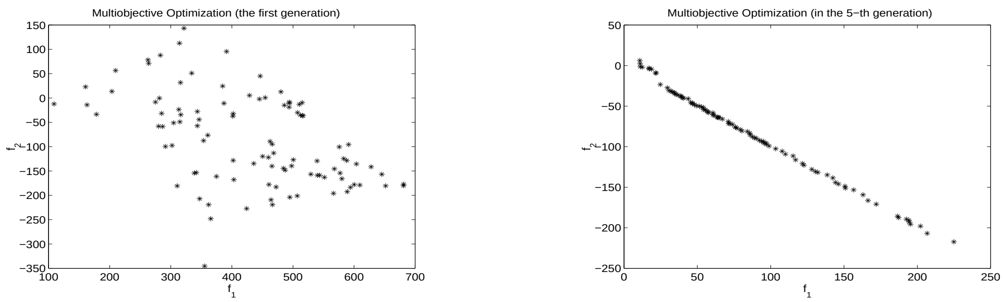

Fig. 1. Pareto optimal solutions for Test Case 1

5.2 Test Case 2

The problem is minimizing $(f_1(\mathbf{x}), f_2(\mathbf{x}))$:

$$f_1(x_1, x_2) = 4x_1^2 + 4x_2^2$$
$$f_2(x_1, x_2) = (x_1 - 5)^2 + (x_2 - 5)^2$$

subject to non-linear constraints:

$$(x_1 - 5)^2 + x_2^2 - 25 \leq 0$$
$$-(x_1 - 8)^2 - (x_2 + 3)^2 + 7.7 \leq 0$$

and bounds: $-15 \leq x_i \leq 30, \ \forall i = 1, 2$.
This problem has the following properties:

- The feasible region is non-convex.
- Some pareto-optimal solutions lie on the boundaries of the feasible region.
- Both objectives are in conflict so that a little reduction of the second objective $\mathbf{f}_2$ leads to a big increase in the first one.

The optimization proccess with an infeasible starting point at $\mathbf{x}_0 = (-10, 30)^T$ is illustrated in *Fig.* **??**. The experimental results with different feasible regions (convex and nonconvex) have shown that MOBES is very robust by searching for the feasible region and very good to maintain a representative sampling of solutions along the pareto-optimal surface.

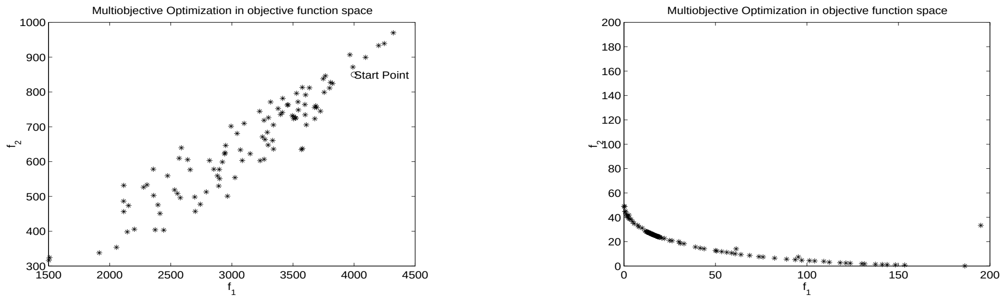

Fig. 2. Optimization process for Test Case 2

6 Conclusion

In this paper the new evolution strategy for multiobjective optimization problems subject to linear and nonlinear constraints is proposed. In opposition to traditional ESs the MOBES allows to start the optimization even from an infeasible point. It is practically useful in many optimization cases where a priori knowledge about the structure of the feasible region is missed. Experiments on some multiobjective optimization problems indicated that the MOBES is robust and give good performances by effectively handling (in)feasible individuals and by maintaining a representative sampling of solutions on the pareto-optimal surface.

References

1. T. Binh. Eine Entwurfsstrategie für Mehrgrößensysteme zur Polgebietsvorgabe. PhD thesis, Institute of Automation, University of Magdeburg, Germany, 1994.
2. T. Binh and U. Korn. An evolution strategy for the multiobjective optimization. The 2^{nd} International Conference Mendel96, Brno , Juni 1996, pages 23–28, 1996.
3. C. Fonseca and P. Fleming. *Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms I: A Unified Formulation.* Technical report, Dept. Automatic Control and Systems Eng., University of Sheffield, U.K., Jan. 1995.
4. D. Goldberg. Genetic algorithms in Search, Optimization, and Machine Learning. Addision–Wesley Publishing Company, Inc., New York, England, Bonn, Tokyo, 1. edition, 1989.
5. J. Horn, N. Nafpliotis, and D. E. Golberg. A niched pareto genetic algorithm for multiobjective optimization. In Proceedings of the 1^{st} IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Interlligence (ICEC'94), 1, 1994.
6. J. Kahlert. Vektorielle Optimierung mit Evolutionsstrategien und Anwendung in der Regelungstechnik. Forschungsbericht VDI Reihe 8 Nr.234, 1991.
7. Z. Michalewicz. Heuristic methods for evolutionary computation techniques. Journal of Heuristics 1(2), pages 177–206, 1995.
8. A. Osyczka and H. Tamura. Pareto set distribution method for multicriteria optimization using genetic algorithm. The 2^{nd} International Conference Mendel96 in Brno, Juni 1996, pages 97–102, 1996.
9. J. Richardson, M. Palmer, G. Liepins, and M. Hilliard. Some guidelines for genetic algorithms with penalty functions. Proceedings of the 3^{rd} International Conference on Genetic Algorithms, Los Altos, CA, Morgan Kaufmann Publishers, pages 191–197, 1989.
10. J. D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In J. Grefenstte, editor, Proceedings of the International Conference on Genetic Algorithms and their Aplication, pages 93–100, 1985.