An Evolution Strategy for Multiobjective Optimization

Lino Costa and Pedro Oliveira
Departamento de Produção e Sistemas
Escola de Engenharia, Universidade do Minho
Campus de Gualtar, 4710 Braga, Portugal
{lac,pno}@dps.uminho.pt

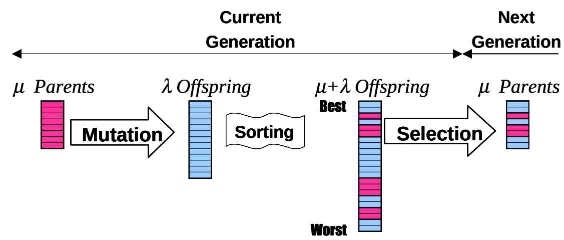
Abstract - Almost all approaches to multiobjective optimization are based on Genetic Algorithms, and implementations based on Evolution Strategies (ESs) are very rare. In this paper, a new approach to multiobjective optimization, based on ESs, is presented. The comparisons with other algorithms indicate a good performance of the Multiobjective Elitist Evolution Strategy.

I. INTRODUCTION

Solving multiobjective engineering problems is a very difficult task due to, in general, for this class of problems, the objectives conflict across a high-dimensional problem space. Thus, the interaction between the multiple objectives gives rise to a set of efficient solutions, known as the Pareto-optimal solutions. During the past decade, the application of evolutionary algorithms to multiobjective optimization has been investigated by several authors, such as Schaffer [10], Fonseca and Fleming [5], Horn et al. [4], Srinivas and Deb [12] and Zitzler and Thiele [14]. Almost all approaches are based on Genetic Algorithms (GAs) [2] which were extended in order to track multiobjective problems. On the other hand, implementations based on Evolution Strategies (ESs) [8] are very rare, such as the algorithm proposed by Knowles and Corne [7]. However, the latter approach does not use some traditional features of ESs, namely, the real coding of decision variables and the adaptation of step sizes for mutation. Thus, it is crucial to investigate how to extend ESs to multiobjective optimization, since, in the past, they prove to be powerful single objective optimizers. In this paper, a new approach to multiobjective optimization, based on ESs, is presented. In the new algorithm, an effort was made in order to maintain the main features of traditional ESs as single objective optimizers. Several mechanisms, like elitism, have been introduced in order to improve the algorithm performance, as previously suggested by Zitzler et al. [15] and Van Veldhuizen and Lamont [13].

In section 2, a short introduction to ES is presented. Section 3 describes the Multiobjective Elitist Evolution Strategy (MEES) implemented. Next, the results of the application to several problems are presented. Finally, some conclusions and future work are addressed.

Figure I
The $(\mu + \lambda)$ Evolution Strategy



II. EVOLUTION STRATEGIES

Evolution Strategies are search procedures that mimic the natural evolution of the species in the natural systems. They were first reported by Rechenberg [8][9] and later by Schwefel [11]. ESs were developed to solve single objective optimization problems. Like GAs, they work with populations of candidate solutions, requiring only data based on the objective function and constraints, and no derivatives or other auxiliary knowledge. However, ESs work directly with the real representation of the decision variables and the transitions rules are deterministic (in particular, selection is a deterministic procedure). In spite of, traditionally, the search of new points was based on one single operator, the mutation operator, more recently, a recombination operator was introduced. One of the most promising features of ESs is that they use adaptive step sizes for mutation.

Figure I illustrates the $(\mu+\lambda)$-ES. The (μ, λ)-ES is similar differing, basically, on the selection procedure. Thus, in $(\mu + \lambda)$-ES, at a given generation, there are μ parents, and λ offspring generated by mutation. Mutation creates new points by adding random normal distributed quantities with mean zero and variance σ_i^2. It is important to note that, for each decision variable, an individual standard deviation σ_i is used (controlling the step sizes). Then, the $\mu+\lambda$ members are sorted according to their objective function values. Finally, the best μ of all the $\mu+\lambda$ members become the parents of the next generation (i.e., the selection takes place between the $\mu+\lambda$ members). On the other hand, in (μ, λ)-ES, the μ best of the λ members generated become the parents of the next generation (i.e., the selection takes place between the λ members).

For many problems, $\lambda/\mu \approx 7$ is suggested. During the search, the step sizes for mutation are adapted. Several self-adaptation schemes are possible. One possibility is to actualize the standard deviations σ_i (for each decision variable) according to the equation:

$$\sigma_i^{(k+1)} = \sigma_i^{(k)} e^{z_i} e^{z} \tag{1}$$

where $z_i \sim N(0, \Delta\sigma^2)$, $z \sim N(0, \Delta\sigma^{`2})$ and $\Delta\sigma$ and $\Delta\sigma^{`}$ are parameters of the algorithm.

Schwefel [11] has reported a remarkable acceleration in the search process, as well as, the facilitation of self-adaptation of parameters by introducing a recombination operator. Basically, the recombination operator consists on, before mutation, to recombine a set of chosen parents to find a new solution. A given number ρ ($1 \leq \rho \leq \mu$) of parents are randomly chosen for recombination. When $\rho = 1$ then there is no recombination. Thus, the nomenclature for ESs can now be extended, and ESs with recombination are usually referred as $(\mu/\rho+\lambda)$-ES or $(\mu/\rho, \lambda)$-ES. Two types of recombination are, mainly, considered: intermediate and discrete recombination. In the intermediate recombination, the components of the offspring are obtained by calculating the average of the corresponding components of parents (randomly selected from the population). In the discrete recombination, each component of the offspring is chosen from one of the ρ parents at random. This procedure allows different combinations of the values of the decision variables from existing solutions in the population.

III. A MULTIOBJECTIVE ELITIST EVOLUTION STRATEGY

The MEES approach to multiobjective optimization differs from conventional ESs with respect to the selection operator emphasizing the non-domination of solutions. Non-domination is tested at each generation in the selection phase, thus defining an approximation to the Pareto optimal set. On the other hand, a sharing method is used to distribute the solutions in the population over the Pareto-optimal region. The usual deterministic selection was also modified in order to track multiobjective optimization. The real representation of the decision variables, mutation and recombination operators remain as usual. The step sizes for mutation were adapted with a non-isotropic self-adaptation scheme as in equation 1.

A. Fitness Assignment

For each generation, all non-dominated solutions of the λ or $\mu + \lambda$ solutions will constitute the 1st front. To these solutions a fitness value of 1 is assigned. In order to maintain diversity, a sharing scheme is then applied to the fitness values of these solutions [1]. Thus, the fitness value of each solution is divided by a quantity, called niche count, proportional to the number of solutions having a distance inferior to a parameter, the σ_{share}. All distances are measured in objective space. Thereafter, the solutions of the 1st front are ignored temporarily, and the rest of solutions are processed. To the second level of non-dominated solutions is assigned a fitness value equal to 1 plus the worst computed fitness value from the solutions in 1st front. Next, the fitness value of each solution in the 2nd front is divided by the respective niche count value. This process is repeated till all the λ or $\mu+\lambda$ solutions are assigned a fitness value. This fitness assignment process will emphasize the non-domination of solutions, since the fitness values of all solutions in the 1st front will have a value inferior to all the fitness values of solutions in the 2nd front, and so on. Moreover, the co-existence of multiple non-dominated solutions is encouraged by the sharing scheme.

B. Selection Operator

In the simplest form, at each generation, only μ from the λ or $\mu+\lambda$ solutions are selected for next generation. Two situations were considered:

- if the number of solutions in 1st front, n_1, is not greater than μ, then a deterministic selection is performed;
- Otherwise, if n_1 is greater than μ, then a tournament selection is performed.

The deterministic selection consists on, after sorting the λ or $\mu + \lambda$ offspring according to their fitness values, to select the μ best (the ones with lower fitness values). This selection is obviously similar to the traditional selection of ESs, in the sense that only the best individuals will be present on the next generation. On the other hand, when the number of solutions in the 1st front is high (greater than μ) then a selection scheme guaranteing that all non-dominated solutions have a possibility of being present in the next generation is adopted. This selection consists on, after sorting the λ or $\mu + \lambda$ offspring, performing a tournament between solutions of the 1st front. The tournament consists on picking two individuals from the offspring and then the best one is selected.

C. Elitist Scheme

The elitist technique is based on a separate population, the secondary population (SP) composed of all (or a part of) potential Pareto optimal solutions found so far during the search process. In this sense, SP is completely independent of the main population and, at the end of the entire search, it contains the set of all non-dominated solutions generated so far.

A parameter θ is introduced in order to control the elitism level. This parameter states the maximum number of non-dominated solutions of SP, the so-called elite, that will be introduced in main population. These non-dominated solutions will effectively participate in the search process. If the number of solutions in SP (n_{SP}) is greater or equal than θ, then θ non-dominated solutions are randomly selected from SP to constitute the elite. Otherwise, only n_{SP} non-dominated solutions are selected from SP to constitute the elite. In the latter case, the elite will only have n_{SP} members.

In its simplest form, for all generations, the new potential Pareto optimal solutions found are stored in SP. The SP update implies the determination of Pareto optimality of all solution stored so far, in order to eliminate those that became dominated. As the size of SP grows, the time to complete this operation may become significant. So, in order to prevent the growing computation times, in general, a maximum SP size is imposed. Thus, the algorithm consists on, for all generations, to store, in SP, each Pareto optimal solution x_{nd} found in the main population if:

1. all solutions in SP are different of x_{nd};
2. none of the solutions in SP dominates x_{nd}.

Next, all solutions in SP that became dominated are eliminated. As mentioned, as the size of SP increases, the execution time and memory requirements also increase. So, it is convenient to keep relatively small sizes of SP. In this sense, the previous algorithm can be modified accordingly. A new parameter d is introduced, stating the minimum desirable distance in objective space between potential Pareto optimal solutions in SP. So, the algorithm is modified by the introduction of the following step:

3. the distance from x_{nd} to any of the non-dominated solutions in SP is greater than d (euclidean distance measured on objective space).

IV. RESULTS

Several experiments were carried out in order to study the effect of the parameters of the elitist scheme, as well as, to compare its performance with some other evolutionary multiobjective approaches.

A. Test Problems

The multiobjective problems were chosen from Zitzler et al. [15]. All problems have two objective functions, no constraints and the Pareto-optimal solutions are known. Table I describes these problems, showing the number of variables and their bounds.

TABLE I
MULTIOBJECTIVE PROBLEMS

Problem	Objective functions
ZDT1 ($n = 30$) $x_i \in [0, 1]$ $i = 1, ..., n$	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^{n} x_i)/(n-1)$
ZDT2 ($n = 30$) $x_i \in [0, 1]$ $i = 1, ..., n$	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^{n} x_i)/(n-1)$
ZDT3 ($n = 30$) $x_i \in [0, 1]$ $i = 1, ..., n$	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)} - \frac{x_i}{g(x)} \sin(10\pi x_1)]$ $g(x) = 1 + 9(\sum_{i=2}^{n} x_i)/(n-1)$
ZDT4 ($n = 10$) $x_1 \in [0, 1]$ $x_i \in [-5, 5]$ $i = 2, ..., n$	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}[x_i^2 - 10\cos(4\pi x_i)]$
ZDT6 ($n = 10$) $x_i \in [0, 1]$ $i = 1, ..., n$	$f_1(x) = 1 - \exp(-4x_1)\sin^6(4\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[(\sum_{i=2}^{n} x_i)/(n-1)]^{0.25}$

The MEES was applied to each problem with a reasonable set of values for the parameters (no effort was made in finding the best parameter setting for each problem). The initial values for standard deviations (step sizes) and parameters for its self adaptation during the search were the suggested for ESs in single objective optimization. The points in the initial population were generated randomly. Several scenarios were considered in order to study the effects of the recombination operator, the selection mechanism, the elitism and d parameter. Thus, for each scenario all parameters values were kept constant except the feature under study (interaction between parameters was not studied in this phase).

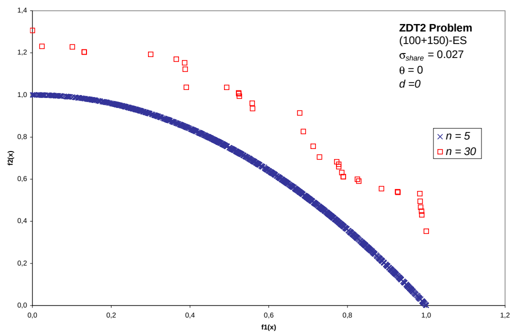
B. Metrics of Performance

Comparing different multiobjective optimization algorithms is substantially more complex than for the case of single objective optimizers, because the optimization goal itself consists on finding a non-dominated set of solutions that is:

- a good approximation to the true Pareto optimal set (the distance between the approximation and the true sets should be minimized);
- a well distributed set in the objective space.

Several attempts can be found in literature to express the above statements by means of quantitative metrics. The metric here considered is described by Knowles and Corne [7] and is based on a statistical method proposed by Fonseca and Fleming [6]. For several executions of the algorithms, a statistical test based on the Mann-Whitney

Figure II
Results for ZDT2 problem with 5 and 30 variables



rank-sum test is applied to the previous collected data. The results of a comparison can be presented in a pair $[a, b]$, where a is the percentage of the objective space on which algorithm A was found statistically superior to B, and b gives the similar percentage for algorithm B. Thus, a is the percentage of the objective space where algorithm A is 'unbeaten' and, b is the percentage of the objective space where algorithm B is 'unbeaten'. So, typically, if $a \approx b \approx 100\%$ then the algorithms A and B have similar results. For all results presented in the paper the statistical significance is at the 5% level and 1000 sampling lines were used.

C. Influence of Recombination

The MEES without the recombination operator seems to have difficulties in obtaining a well distributed set of non-dominated solutions when applied to multiobjective problems with a high number of variables. This is illustrated by Figure II, which represents the non-dominated solutions obtained, in one single run, for ZDT2 problem with 5 and 30 variables for an (100+150)-ES without any recombination and, with σ_{share}=0.027, $d = 0$ and $\theta = 0$. The stopping criterion was to terminate the execution after 250 generations. It is clear that a good definition of the approximation to the Pareto-optimal set was obtained for the ZDT2 problem with 5 variables. However, for 30 variables, the results are poor, in the sense, that the solutions are far from the true Pareto-optimal front and, they are not uniformly distributed in the objective space. Since MEES without recombination seems to perform poorly for large dimensional multiobjective problems, several scenarios of MEES with recombination were tested. Scenarios that combine the most popular recombination schemes were considered:

- without any recombination (NOrec scenario);
- intermediate recombination on variables and standard deviations (IIrec scenario);
- intermediate recombination on variables and discrete recombination on standard deviations (IDrec scenario);
- discrete recombination on variables and intermediate recombination on standard deviations (DIrec scenario);
- discrete recombination on variables and standard deviations (DDrec scenario).

For scenarios with recombination, an (100/100,250)-ES was applied (obviously, an (100,250)-ES was considered when no recombination exists) with σ_{share}=0.027, $d = 0$ and $\theta = 0$. As before, the stopping criterion was to terminate the execution after 250 generations. For each scenario, the MEES was executed 30 times. Table II presents the results obtained for all scenarios for the ZDT1 problem. All scenarios were compared in pairs using the statistical technique as previously described. It is clear that the best results were obtained for DDrec scenario, i.e., when discrete recombination is applied to decision variables and standard deviations. The MEES with discrete recombination on variables and standard deviations (an (100/100,250)-ES) can now be compared with the performance of MEES without any recombination for the ZDT2 problem. The comparison is illustrated by Figure III, which represents the non-dominated solutions obtained in one single run after 250 generations. The approximation to the Pareto-optimal front obtained with the MEES with recombination, was far better than the obtained without any recombination.

TABLE II
INFLUENCE OF RECOMBINATION (ZDT1 PROBLEM)

ZDT1	IIrec	IDrec	DIrec	DDrec
NOrec	[100,3.8]	[100,2.2]	[2.7,100]	[2.7,100]
IIrec	-	[100,7.2]	[3.6,100]	[3.7,100]
IDrec	-	-	[2.1,100]	[2.1,100]
DIrec	-	-	-	[56.7,100]

Figure III
Results for ZDT2 problem with and without recombination

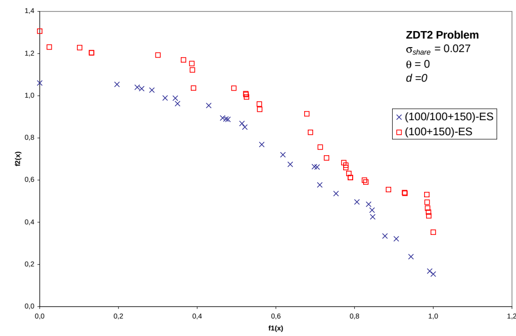
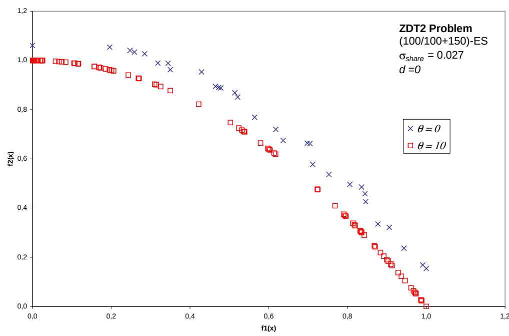


TABLE III
INFLUENCE OF ELITISM (ZDT1 PROBLEM)

ZDT1	$\theta = 10$	$\theta = 20$	$\theta = 50$	$\theta = 100$
$\theta = 0$	[0.3,100]	[0.3,100]	[0.3,100]	[0.3,100]
$\theta = 10$	-	[100,32.1]	[100,13.5]	[100,13.5]
$\theta = 20$	-	-	[96.9,83.9]	[96.9,83.9]
$\theta = 50$	-	-	-	[100,100]

Figure IV
Results for ZDT2 problem for $\theta = 0$ and $\theta = 10$



D. Influence of Elitism

In order to study the influence of the elitism level, an (100/100,250)-ES with discrete recombination on variables and standard deviations was applied to the ZDT1 problem. The same values for the parameters were considered with the exception of θ, which was varied from 0 to 100 as in Table III. The d parameter was fixed equal to 0 in order to guarantee that in SP all non-dominated solutions found during the search are present. This table shows that for increasing values of θ there is a degradation of the performance of the algorithm, due to the lack of diversity in main population. However, it is also clear that, for the values of θ tested, the best results were obtained with elitism. Furthermore, consistently, the best results were obtained with $\theta = 10$. The comparison between different levels of elitism is illustrated by Figure IV, which represents the non-dominated solutions obtained in one single run, after 250 generations for the ZDT2 problem, with $\theta = 0$ and $\theta = 10$. It is clear that the approximation to the Pareto-optimal front obtained with $\theta = 10$ was far better than with $\theta = 0$.

E. Comparison with other algorithms

The elitist ES was compared with four algorithms for the test problems (ZDT1 to ZDT6 problems). These results were published by Ziztler et al. [15]. The algorithms considered here are:

- HLGA: Hajela and Lin's weighted-sum based approach [3];
- VEGA: Vector Evaluated Genetic Algorithm [10];

TABLE IV
COMPARISON BETWEEN ALGORITHMS (ZDT1 PROBLEM)

ZDT1	HLGA	VEGA	NSGA	SPEA	$MEES_{10}$
$MEES_0$	[100,8.3]	[100,11.4]	[100,1.3]	[1.7,100]	[1.7,100]
HLGA	-	[75.2,77.5]	[8.8,100]	[8.3,100]	[8.3,100]
VEGA	-	-	[12.0,100]	[11.3,100]	[11.2,100]
NSGA	-	-	-	[1.3,100]	[1.3,100]
SPEA	-	-	-	-	[1.9,100]

TABLE V
COMPARISON BETWEEN ALGORITHMS (ZDT2 PROBLEM)

ZDT2	HLGA	VEGA	NSGA	SPEA	$MEES_{10}$
$MEES_0$	[100,16.3]	[100,3.3]	[100,3.5]	[17.8,97.3]	[1.5,100]
HLGA	-	[25.7,100]	[19.0,100]	[16.5,100]	[15.9,100]
VEGA	-	-	[4.6,100]	[3.4,100]	[3.2,100]
NSGA	-	-	-	[3.6,100]	[3.4,100]
SPEA	-	-	-	-	[1.9,100]

- NSGA: Nondominated Sorting Genetic Algorithm [12];
- SPEA: Strength Pareto Evolutionary Algorithm [14].

For MEES, an (100/100,150)-ES with discrete recombination in variables and standard deviations was considered. The MEES was applied without and with elitism ($MEES_0$ and $MEES_{10}$, respectively). The d and σ_{share} parameters were fixed equal to 0 and 0.027, respectively. The stopping criterion was to terminate the search after 100 generations. As described with more detail in [15], for algorithms HLGA, VEGA, NSGA and SPEA, the population size was 100 (for SPEA the population size was 80 with an external non-dominated set of 20 points). The crossover and mutation rates were 0.8 and 0.01, respectively. The maximum number of generations was 250. The niching parameter was fixed in 0.48862. All algorithms were executed 30 times for each test problem and, for each run, the set of all non-dominated solutions generated during the entire search was taken as the outcome of one optimization run (off-line performance). The number of objective function evaluations was the same for algorithms HLGA, VEGA, NSGA and SPEA (approximately, 25000 evaluations). The number of objective function evaluations required by MEES was inferior than the other algorithms (approximately, 15000 evaluations). Tables IV to VIII present the results of compari-

TABLE VI
COMPARISON BETWEEN ALGORITHMS (ZDT3 PROBLEM)

ZDT3	HLGA	VEGA	NSGA	SPEA	$MEES_{10}$
$MEES_0$	[100,13.7]	[100,6.6]	[100,2.4]	[2.4,100]	[2.7,100]
HLGA	-	[55.9,82.9]	[14.3,100]	[13.7,100]	[15.6,100]
VEGA	-	-	[6.9,100]	[6.6,100]	[7.3,100]
NSGA	-	-	-	[2.4,100]	[2.7,100]
SPEA	-	-	-	-	[0.9,100]

TABLE VII
COMPARISON BETWEEN ALGORITHMS (ZDT4 PROBLEM)

ZDT4	HLGA	VEGA	NSGA	SPEA	$MEES_{10}$
$MEES_0$	[100,32.7]	[100,9.8]	[100,16.7]	[100,13.7]	[100,27.9]
HLGA	-	[87.6,47.4]	[33.1,100]	[33.1,100]	[32.7,100]
VEGA	-	-	[10.6,100]	[10.6,100]	[9.8,100]
NSGA	-	-	-	[82.2,100]	[16.7,100]
SPEA	-	-	-	-	[13.7,100]

TABLE VIII
COMPARISON BETWEEN ALGORITHMS (ZDT6 PROBLEM)

ZDT6	HLGA	VEGA	NSGA	SPEA	$MEES_{10}$
$MEES_0$	[63.8,36.9]	[64.6,35.7]	[58.8,41.5]	[51.3,49.2]	[85.1,47.0]
HLGA	-	[100,11.2]	[4.9,100]	[3.5,100]	[25.0,76.0]
VEGA	-	-	[0.0,100]	[0.0,100]	[34.8,65.4]
NSGA	-	-	-	[5.2,100]	[15.5,85.3]
SPEA	-	-	-	-	[17.1,85.4]

TABLE IX
GLOBAL PERFORMANCE

	HLGA	VEGA	NSGA	$MEES_0$	SPEA	$MEES_{10}$
ZDT1	8.8	11.4	1.3	100	100	100
ZDT2	16.5	3.4	3.6	100	100	100
ZDT3	14.3	6.9	2.7	100	100	100
ZDT4	32.7	10.6	82.2	100	100	100
ZDT6	25.0	11.2	41.5	63.8	100	76.0

son of these algorithms with MEES. Two classes of algorithms can be distinguished, those that do not use elitism (HLGA, VEGA NSGA and $MEES_0$) and, those that use, explicitly, elitism in the search (SPEA and $MEES_{10}$) . Thus, considering only the results obtained with non elitist algorithms, the best results were obtained by $MEES_0$ for all test problems considered. Moreover, $MEES_0$ has outperformed the elitist approaches SPEA in two problems (ZDT4 and ZDT6 problems) and $MEES_{10}$ in one problem (ZDT4 problem). However, $MEES_{10}$ has beaten SPEA in all the problems considered. Table IX resumes the results obtained by different algorithms for all problems. The values in the table, for each algorithm, are the median percentage of the objective space that is 'unbeaten' when compared with the remaining algorithms. From this table, it is clear that, in general, the elitism is useful in guiding the search. The best non elitist algorithm seems to be $MEES_0$. SPEA, globally, outperformed $MEES_{10}$ on ZDT6 problem.

V. CONCLUSIONS AND FUTURE WORK

In this work, a new Elitist Evolution Strategy for multiobjective optimization was presented. This approach incorporates the main features of traditional single objective Evolution Strategies, like real representation of the decision variables and self-adaptation of step sizes.

The algorithm was tested on several test problems in order to investigate the influence of some factors on its performance, as well as, to compare its performance with other multiobjective evolutionary approaches. As expected, the results indicated that recombination and elitism are essential for obtaining good approximations to the Pareto-optimal front. The Multiobjective Elitist Evolution Strategy without elitism ($MEES_0$) outperformed the other non elitist approaches (HLGA, VEGA and NSGA) for all the test problems considered. The results of the Multiobjective Elitist Evolution Strategy with elitism ($MEES_{10}$) and SPEA were similar. It should be noted that the number of function evaluations required by MEES is inferior than the others algorithms being compared.

Future work will concentrate on the study of parameters like population sizes, initial step sizes and self-adaptation schemes. The influence of the parameter that controls the density of points in the approximation set to the Pareto-optimal front will also be investigated.

References

[1] K. Deb and D. Goldberg. "An investigation of niche and species formation in genetic function optimization". Proceedings of the Third International Conference on Genetic Algorithms, pp. 42-50, 1989.

[2] D. Goldberg. Genetic Algorithms in Search,Optimization, and Machine Learning. Addison-Wesley,1989.

[3] P. Hajela and C.-Y. Lin. "Genetic search strategies in multicriterion optimal design". *Structural Optimization*, 4, pp. 99-107, 1992.

[4] J. Horn, N. Nafploitis and D. Goldberg. "A niched Pareto genetic Algorithm for multi-objective optimization". Proceedings of the First IEEE Conference on Evolutionary Computation, pp. 82-87, 1994.

[5] C. M. Fonseca and P. J. Fleming. "Genetic Algorithms for multi-objective optimization: Formulation, discussion and generalization". Proceedings of the Fifth International Conference on Genetic Algorithms, S. Forrest, Ed. San Mateo, CA: Morgan Kauffman, pp. 416-423, 1993.

[6] C. M. Fonseca and P. J. Fleming. "On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers" Parallel Problem Solving from Nature IV, H.-M. Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel, Eds Berlin: Germany: Springer, pp. 584-593, 1995.

[7] J. D. Knowles and D. W. Corne. "Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy". *Evolutionary Computation*, Vol. 8(2), pp. 149-172, 2000.

[8] I. Rechenberg. Evolutionsstrategie '94. Frommann-Holzboog, Stuttgart, 1994.

[9] I. Rechenberg. Cybernetic Solution Path of an Experimental Problem. Royal Aircraft Establishment, Library Translation No. 1122, Farnborough, England, 1964.

[10] J. D. Schaffer. "Multiple objective optimization with vector evaluated genetic algorithms". Proceedings of the First International Conference on Genetic Algorithms, J.J Grefensttete, Ed. Hillsdale, pp.93-100, 1985.

[11] H-P. Schwefel. Evolution and Optimal Seeking. John Wiley and Sons, 1995.

[12] N. Srinivas and K. Deb. "Multi-Objective function optimization using non-dominated sorting genetic algorithms". *Evolutionary Computation*, Vol. 2, pp. 221-248, 1995.

[13] D. Van Veldhuizen and G. Lamont. "MultiObjective Evolutionary Algorithms: Analysing the State-of-the-Art". *Evolutionary Computation*, Vol. 8(2), pp. 125-147, 2000.

[14] E. Zitzler and L. Thiele. "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach". *IEEE Transactions on Evolutionary Computation*, 3(4), pp. 257-271, 1999.

[15] E. Zitzler, K. Deb and L. Thiele. "Comparison of multiobjective evolutionary algorithms: Empirical results". *Evolutionary Computation*, Vol. 8, pp. 173-195, 2000.