Agent-Based Co-Operative Co-Evolutionary Algorithm for Multi-Objective Optimization

Rafał Dreżewski, Leszek Siwik

Department of Computer Science
AGH University of Science and Technology, Kraków, Poland
{drezew,siwik}@agh.edu.pl

Abstract. Co-evolutionary algorithms are a special type of evolutionary algorithms, in which the fitness of each individual depends on other individuals' fitness. Such algorithms are applicable in the case of problems for which the formulation of explicit fitness function is difficult or impossible. Co-evolutionary algorithms also maintain population diversity better than "classical" evolutionary algorithms. In this paper the agent-based version of co-operative co-evolutionary algorithm is presented and applied to multi-objective test problems. The proposed technique is also compared to two "classical" multi-objective evolutionary algorithms.

1 Introduction

Co-evolutionary algorithms (CEAs) [20] are a special type of *evolutionary algorithms (EAs)* [2], in which fitness of the given individual depends on fitness of other individuals existing in the population. Such algorithms have some interesting features, among others the possibility of application in the case of problems for which formulation of fitness function is difficult or impossible, maintaining the population diversity, "arms races", and so on. Co-operative co-evolutionary algorithms [21] are CEAs in which there exist several sub-populations, and each of them solves only part of the given problem—the whole solution is represented by the group of individuals composed of representants of all sub-populations. Co-operative co-evolutionary approach was applied to multi-objective optimization by Iorio and Lee [14].

Evolutionary multi-agent systems (EMAS) represent agent-based approach to evolutionary computations. In such systems the population of agents evolve—agents can reproduce, die, compete for resources, observe the environment, communicate with other agents, and make autonomously all their decisions. All these features lead to completely decentralized evolutionary processes. *Co-evolutionary multi-agent systems (CoEMAS)* additionally allow us to define interactions between species of agents. This type of systems have been already applied to multi-modal optimization [7] and multi-objective optimization [8, 9].

In this paper the co-operative co-evolutionary multi-agent system for multi-objective optimization (CCoEMAS) is presented. The proposed system is evaluated with the use of test problems proposed by Zitzler ([28]) and the results are compared to those of two "classical" multi-objective evolutionary algorithms.

2 Evolutionary Multi-Objective Optimization

During most real-life decision processes a lot of different (often contradictory) factors have to be considered, and the decision maker has to deal with an ambiguous situation: the solutions which optimize one criterion may prove insufficiently good considering the others. From the mathematical point of view such multi-objective (or multi-criteria) problem can be formulated as follows [1, 28, 26].

Let the problem variables be represented by a real-valued vector:

$$\overrightarrow{x} = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N \tag{1}$$

where N is the number of variables. Then a subset of $\mathbb{R}^N$ of all possible (feasible) decision alternatives (options) can be defined by a system of:

- inequalities (constraints): $g_k(\overrightarrow{x}) \geq 0$ and $k = 1, 2, \ldots, K$,
- equalities (bounds): $h_l(\overrightarrow{x}) = 0$, $l = 1, 2, \ldots, L$

and denoted by $\mathcal{D}$. The alternatives are evaluated by a system of M functions (objectives) denoted here by vector $F = [f_1, f_2, \ldots, f_M]^T$:

$$f_m : \ \mathbb{R}^N \rightarrow \mathbb{R}, \quad m = 1, 2, \ldots, M \tag{2}$$

The key issue of optimality in the Pareto sense is the *weak domination relation.* Alternative $\overrightarrow{x}^a$ is dominated by $\overrightarrow{x}^b$ (which is often denoted by $\overrightarrow{x}^b \succeq \overrightarrow{x}^a$) if and only if (assuming maximization of all the objectives):

$$\forall m \;\; f_m(\overrightarrow{x}^a) \leq f_m(\overrightarrow{x}^b) \text{ and } \exists m \;\; f_m(\overrightarrow{x}^a) < f_m(\overrightarrow{x}^b) \tag{3}$$

A solution in the Pareto sense of the multi-objective optimization problem means determination of all non-dominated (in the sense of the defined above *weak domination relation*) alternatives from the set $\mathcal{D}$, which is sometimes called a *Pareto set*:

$$\mathcal{P} = \{ \overrightarrow{x} \in \mathcal{D} \;\; | \;\; \neg \exists \overrightarrow{x}^a \in \mathcal{D} \;\; \overrightarrow{x}^a \succeq \overrightarrow{x} \} \tag{4}$$

At the same time the non-dominated alternatives create in criteria space a set called *Pareto frontier*:

$$\mathcal{PF} = \{ \overrightarrow{y} = F(\overrightarrow{x}) \in \mathbb{R}^M \;\; | \;\; \overrightarrow{x} \in \mathcal{P} \} \tag{5}$$

Unfortunately, when searching for the approximation of the Pareto frontier in the whole, classical computational methods often prove ineffective for many (real) decision problems. The corresponding models are too complex or the formulas applied too complicated, or it can even occur that some formulations must be rejected in the face of numerical instability of available solvers. That is why so much attention is paid to methods based on evolutionary algorithms. These methods are relatively insensitive to complexity of the problem and give the approximation of the whole Pareto frontier with controllable adequacy, which means that a solving process can be stopped by a decision maker anytime he is satisfied.

For the last 20 years a variety of evolutionary multi-criteria optimization techniques have been proposed [4, 18, 24, 27, 25]. In the Deb's typology of evolutionary multi-objective algorithms (EMOAs) firstly the elitist and non-elitist ones are distinguished[1] [5]. Each of these groups include many practically used algorithms such as:

- elitist EMOAs: Rudolph's algorithm [22], distance-based Pareto GA [19], strength Pareto EA [29], multi-objective micro GA [3], Pareto-archived evolution strategy [15], multi-objective messy GA [26], etc.
- non-elitist EMOAs: vector-optimized evolution strategy [16], random weighted GA [17], weight-based GA [11], niched-pareto GA [13], non-dominated sorting GA [23], multiple objective GA [10], distributed sharing GA [12] etc.

The main difference between these two groups of techniques consists in utilizing the so-called elite-preserving operators that give the best individuals (the elite of the population) the opportunity to be directly carried over to the next generation regardless of the actual selection mechanism used. Of course, if the algorithm finds a better solution than the one in the elite, this solution becomes a new elitist solution.

3 Co-Operative Co-Evolutionary Multi-Agent System

The functioning principles of the system presented in this paper are in accordance with the general model of co-evolution in multi-agent system proposed in [6]. The formal model of the co-evolutionary multi-agent system for multi-objective optimization may be found for example in [9].

CoEMAS systems are composed of the environment, which has usually the structure of graph with every node connected with its four neighbors, and agents, which "live" within the environment. Agents can reproduce, die, and compete for limited resources. Because of decentralized nature of CoEMAS systems there is no possibility of applying selection mechanisms known from "classical" evolutionary algorithms. Resources defined within the system play the role of decentralized selection mechanism: agents need resources to perform every action (like reproduction, migration, and so on). In order to use the resources as a selection mechanism, they are transferred from "worse" to "better" agents ("better" in the sense of solution of the given problem encoded within the agent's genotype). Usually there is only one resource defined. The total amount of the resource within the system is constant, what means that there is closed circulation of the resource.

In CoEMAS systems different co-evolutionary mechanisms can be defined: predator-prey, host-parasite, co-operative interactions, sexual selection mecha-

[1] Deb's typology includes also so-called *constrained EMOAs*—techniques that support handling constraints

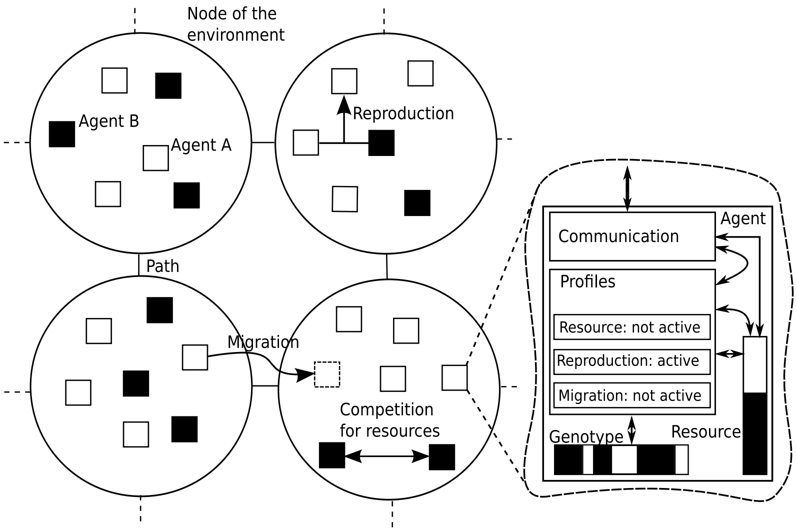

Fig. 1. Co-operative co-evolutionary multi-agent system

nism, and so on. In the system presented in this paper the co-operative interactions are used. There are several sub-populations (species) within the co-operative co-evolutionary system used in experiments (see fig.1). One criteria is assigned to each species (it is used in order to evaluate the agents), so the number of species corresponds to the number of criteria of the given problem. Agents compete for resources only within the species—there is no competition between agents that belong to different species. Reproduction takes place when the agent has enough resources to perform it. The agent searches for a reproduction partner within the node in which it is located—the partner must come from one of the opposite species. In the system real-valued vectors are used as agents' genotypes. Mutation with self-adaptation and intermediate recombination are used as evolutionary operators [2]. Two agents produce one offspring and give it some of their resources.

The very important mechanism is the decision making process performed by the agent. Each agent has several profiles (fig. 1), which are activated in order to realize the agent's goals. Whenever there are some profiles with active goals ("active" means here that they should be realized as soon as possible) the profile with the highest priority is chosen, and its active goal is realized with the use of actions that can be performed within the given profile.

In the CCoEMAS the agent has three profiles: resource (with highest priority), reproduction, and migration. Within the resource profile there are three actions: *seek* (which is performed in order to find the agent that is dominated by the given agent and located within the same node), *get* (which gets resources from dominated agent), and *die* (which is performed when the agent is out of resources—such agent is removed from the system). Within the reproduction profile there are the following actions: *seek* (which is used to search for partner when the amount of resource is above the given level), *clone* (which clones the agent), *rec* (which performs the recombination), *mut* (which performs the mutation), and *give* (which is used in order to give the offspring some of the parent's resources). And finally, within the migration profile there is one action defined: *migr* (it is performed by the agent, which migrates to another node of the environment). The migration takes place when the agent has enough resources.

4 Experimental Results

Agent-based co-evolutionary algorithm that is the subject of considerations in the course of this paper was assessed tentatively using inter alia so called Zitzler problems—ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6 (their definitions can be found in [28]). To assess proposed approach HVR measure ([5]) was used as the metrics measuring both: closeness to the model Pareto frontier and dispersing solutions over the whole frontier. The size of population of algorithm that is being assessed and benchmarking algorithms are as follows: CCoEMAS—200, NSGA2—300 and SPEA—100. In the table 1 there are presented selected values of parameters for co-operative co-evolutionary multi-agent system used during experiments.

Table 1. Selected configuration parameters

Parameter	**Comments**	**Value in CCoEMAS**
InitialResourcesPerSpecimen	Resources possessed initially by individual just after its creation	50
ResourcesToTransfer	Resources transferred in the case of domination	30
MutationProbability	—	0.5
ResourcesForCrossover	Resources required for reproduction	30

In the figures 2–6 there are presented values of HVR measure obtained with time by co-evolutionary multi-agent system with co-operation for ZDT1 (fig. 2), ZDT2 (fig. 3), ZDT3 (fig. 4), ZDT4 (fig. 5) and ZDT6 (fig. 6) problems. There are also presented results obtained by NSGA2 and SPEA2 algorithms.

Co-evolutionary multi-agent system, as not so complex algorithm as NSGA2 or SPEA2, initially allows for obtaining better solutions, but with time classical

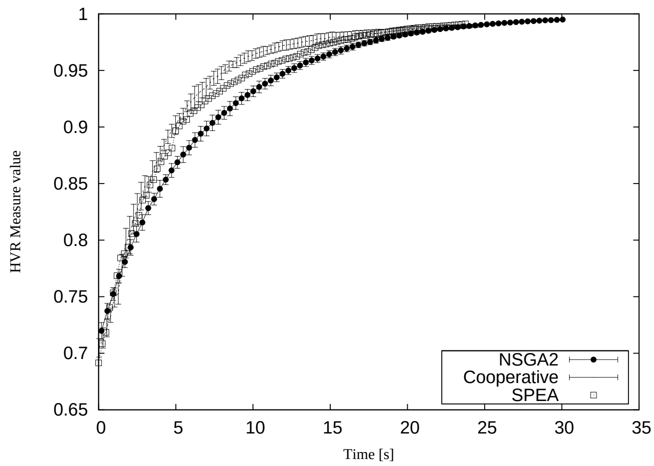

Fig. 2. HVR values obtained by CCoEMAS, SPEA2, and NSGA2 run against Zitzler's ZDT1 problem

algorithms—especially NSGA2—are the better alternatives. It is however worth to mention that in the case of ZDT4 problem this characteristic seems to be reversed—i.e. initially classical algorithms seem to be better alternatives, but finally co-evolutionary multi-agent system with co-operation leads for obtaining better solutions (observed as higher values of HVR metrics). Such characteristics can result from the fact that agent-based evolutionary systems are not so complex algorithms as NSGA2 or SPEA2 algorithms, so—in the consequence—from the time point of view, agent-based algorithms realize more computational steps than "classical algorithms" and they are able to obtain better solutions, but with time classical algorithms perform enough steps to obtain high-quality results.

5 Summary and Conclusions

As it can be observed in section 4 analyzing the quality of obtained solutions as the function of time, co-evolutionary multi-agent system with co-operation is very attractive alternative since initially (in our experiments during c.a. 15 seconds) solutions proposed by this algorithm were better than solutions proposed by "classical" (non agent-based) algorithms. Short explanation of such a phenomenon was given in section 4 and obviously in the function of consecutive steps classical algorithms are absolutely much more effective (but usually, it is not so important for the user/decision maker in how many steps valuable results were obtained but how fast it was done). It is enough to mention that each step of NSGA2 algorithm consists inter alia of ordering population according to the consecutive levels of domination, crowding etc.—it is obviously very effective (i.e. in each step algorithm heads effectively toward model Pareto frontier) but each

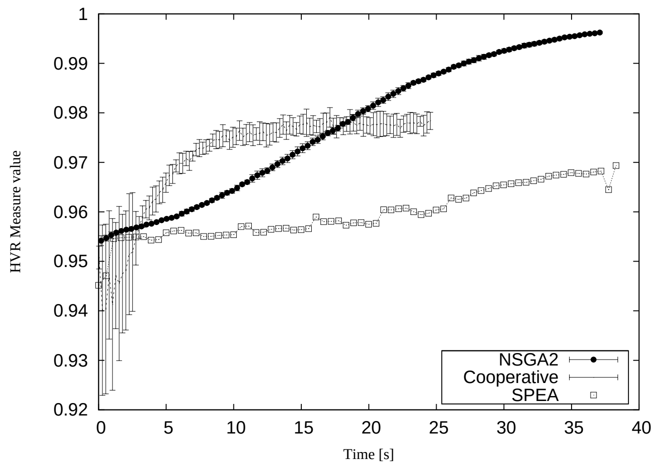

Fig. 3. HVR values obtained by CCoEMAS, SPEA2, and NSGA2 run against Zitzler's ZDT2 problem

step of this algorithm is complex and in the consequence time consuming. So, in the function of algorithm step—classical algorithms are more effective, also in the function of time, final results proposed by NSGA2 are slightly better than results obtained by co-evolutionary multi-agent system, but if time of obtaining valuable results is crucial—agent-based approach seems to be very attractive alternative.

The future plans include further investigation of the proposed mechanism. It could be interesting to modify the co-operation proposed in this paper in such a way that agents from different sub-populations (species) would specialize in different criteria and form aggregates (teams) composed of the representants of different species in order to solve the problem.

References

1. A. Abraham, L. C. Jain, and R. Goldberg. *Evolutionary Multiobjective Optimization Theoretical Advances and Applications*. Springer, 2005.
2. T. Bäck, D. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*. IOP Publishing and Oxford University Press, 1997.
3. C. Coello Coello and G. Toscano. Multiobjective structural optimization using a micro-genetic algorithm. *Structural and Multidisciplinary Optimization*, 30(5):388–403, November 2005.
4. C. A. Coello Coello, D. A. Van Veldhuizen, and G. B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, 2002.
5. K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, 2001.

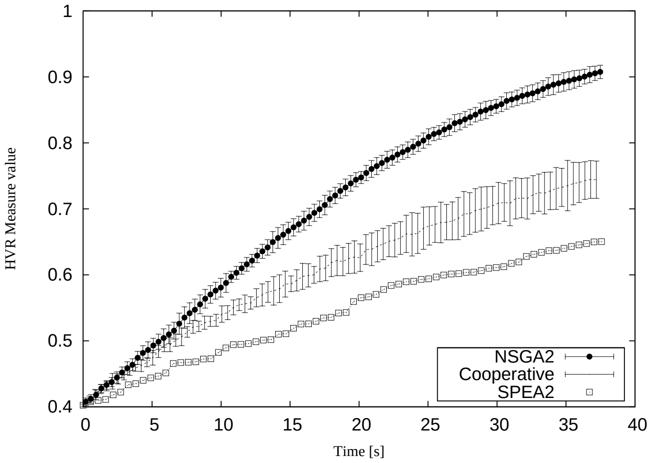

Fig. 4. HVR values obtained by CCoEMAS, SPEA2, and NSGA2 run against Zitzler's ZDT3 problem

6. R. Dreżewski. A model of co-evolution in multi-agent system. In V. Mařík, J. Müller, and M. Pěchouček, editors, *Multi-Agent Systems and Applications III*, volume 2691 of *LNCS*, pages 314–323, Berlin, Heidelberg, 2003. Springer-Verlag.
7. R. Dreżewski. Co-evolutionary multi-agent system with speciation and resource sharing mechanisms. *Computing and Informatics*, 25(4):305–331, 2006.
8. R. Dreżewski and L. Siwik. Co-evolutionary multi-agent system with sexual selection mechanism for multi-objective optimization. In *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI 2006)*. IEEE, 2006.
9. R. Dreżewski and L. Siwik. Multi-objective optimization technique based on co-evolutionary interactions in multi-agent system. In M. Giacobini, editor, *Applications of Evolutionary Computing*, volume 4448 of *LNCS*, pages 179–188. Springer-Verlag, 2007.
10. C. Fonseca and P. Fleming. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In *Genetic Algorithms: Proceedings of the Fifth International Conference*, pages 416–423. Morgan Kaufmann, 1993.
11. P. Hajela, E. Lee, and C. Lin. Genetic algorithms in structural topology optimization. In *Proceedings of the NATO Advanced Research Workshop on Topology Design of Structures*, volume 1, pages 117–133, 1993.
12. T. Hiroyasu, M. Miki, and S. Watanabe. Distributed genetic algorithms with a new sharing approach. In *Proceedings of the Conference on Evolutionary Computation*, volume 1. IEEE Service Center, 1999.
13. J. Horn, N. Nafpliotis, and D. E. Goldberg. A niched pareto genetic algorithm for multiobjective optimization. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, pages 82–87, Piscataway, New Jersey, 1994. IEEE Service Center.
14. A. Iorio and X. Li. A cooperative coevolutionary multiobjective algorithm using non-dominated sorting. In K. Deb, R. Poli, W. Banzhaf, H.-G. Beyer, E. K. Burke, P. J. Darwen, D. Dasgupta, D. Floreano, J. A. Foster, M. Harman, O. Holland, P. L.

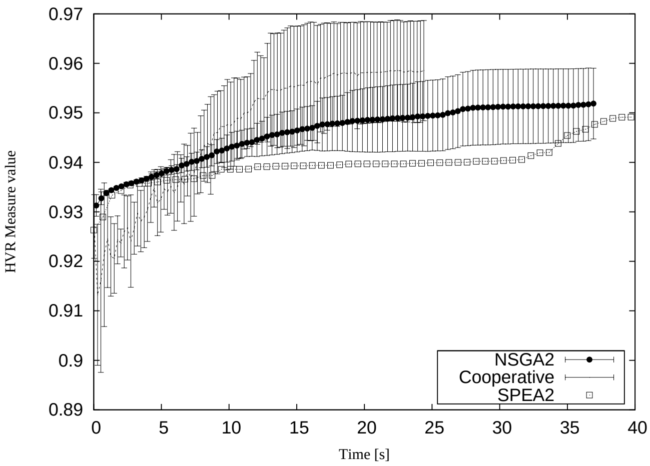

Fig. 5. HVR values obtained by CCoEMAS, SPEA2, and NSGA2 run against Zitzler's ZDT4 problem

Lanzi, L. Spector, A. Tettamanzi, D. Thierens, and A. M. Tyrrell, editors, *Genetic and Evolutionary Computation - GECCO 2004*, volume 3102-3103 of *LNCS*, pages 537–548. Springer-Verlag, 2004.

15. J. Knowles and D. Corne. Approximating the nondominated front using the pareto archived evolution strategy. *Evolutionary Computation*, 8(2):149–172, 2000.
16. F. Kursawe. A variant of evolution strategies for vector optimization. In H. Schwefel and R. Manner, editors, *Parallel Problem Solving from Nature. 1st Workshop, PPSN I*, volume 496, pages 193–197, Berlin, Germany, 1991. Springer-Verlag.
17. T. Murata and H. Ishibuchi. Moga: multi-objective genetic algorithms. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, volume 1, pages 289–294. IEEE, IEEE Service Center, November 1995.
18. A. Osyczka. *Evolutionary Algorithms for Single and Multicriteria Design Optimization.* Physica Verlag, 2002.
19. A. Osyczka and S. Kundu. A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Structural and Multidisciplinary Optimization*, 10(2):94–99, October 1995.
20. J. Paredis. Coevolutionary algorithms. In T. Bäck, D. Fogel, and Z. Michalewicz, editors, *Handbook of Evolutionary Computation, 1st supplement.* IOP Publishing and Oxford University Press, 1998.
21. M. A. Potter and K. A. De Jong. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation*, 8(1):1–29, 2000.
22. G. Rudolph. Evolutionary search under partially ordered finite sets. In M. F. Sebaaly, editor, *Proceedings of the International NAISO Congress on Information Science Innovations (ISI 2001)*, pages 818–822, Dubai, U. A. E., 2001. ICSC Academic Press.
23. N. Srinivas and K. Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3):221–248, 1994.

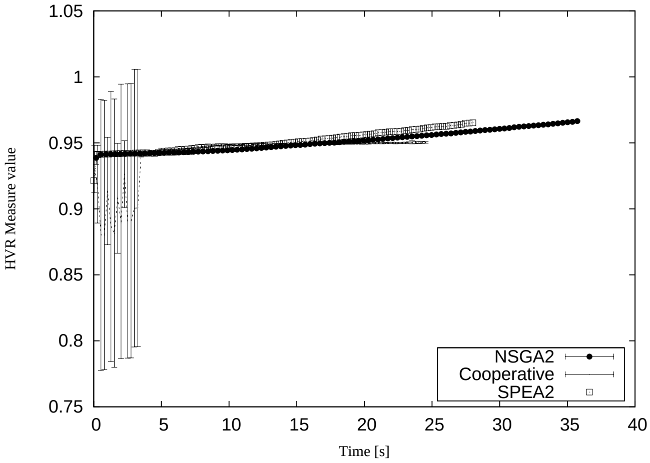

Fig. 6. HVR values obtained by CCoEMAS, SPEA2, and NSGA2 run against Zitzler's ZDT6 problem

24. K. C. Tan, E. F. Khor, and T. H. Lee. *Multiobjective Evolutionary Algorithms and Applications*. Springer, 2005.
25. G. Toscano. *On the Use of Self-Adaptation and Elitism for Multiobjective Particle Swarm Optimization*. PhD thesis, Centro de Investigacion y Estudios Avanzados del Instituto Politecnico Nacional, 2005.
26. D. A. Van Veldhuizen. *Multiobjective Evolutionary Algorithms: Classifications, Analyses and New Innovations*. PhD thesis, Graduate School of Engineering of the Air Force Institute of Technology Air University, 1999.
27. M. A. Villalobos. *Analisis de Heuristicas de Optimizacion para Problemas Multiobjetivo*. PhD thesis, Centro de Investigacion y de Estudios Avanzados del Instituto Politecnico Nacional, 2005.
28. E. Zitzler. *Evolutionary algorithms for multiobjective optimization: methods and applications*. PhD thesis, Swiss Federal Institute of Technology, Zurich, 1999.
29. E. Zitzler and L. Thiele. An evolutionary algorithm for multiobjective optimization: The strength pareto approach. Technical Report 43, Swiss Federal Institute of Technology, Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 1998.