Málaga, 22 de noviembre de 2008

Informe Ejecutivo

TÍTULO: MULTIOBJ-1.2: Un Algoritmo Híbrido Celular basado en Evolución Diferencial (CellDE)

RESUMEN: En este documento se describe CellDE: un nuevo algoritmo multi-objetivo híbirido celular basado en evolución diferencial. Además, para comprobar el rendimiento de CellDE, se presenta una comparativa con cuatro algoritmos que conforman el estado del arte en optimización multi-objetivo: NSGA-II, SPEA2, MOCell y GDE3. Esta comparativa se ha realizado en la base de dos indicadores de calidad (Epsilon (I_ϵ) e Hypervolume (HV)) y un *benchmark* compuesto 16 problemas (DTLZ y WFG) configurados con tres objetivos.

OBJETIVOS:

1. Diseño de un nuevo algoritmo híbrido celular basado en el mecanismo de la evolución diferencial.
2. Resolver problemas con más de dos objetivos.
3. Comparar la calidad de las soluciones obtenidas con los algoritmos de optimización que conforman el estado del arte.

CONCLUSIONES:

1. Se ha diseñado un nuevo algoritmo basado en la hibridación de MOCell y GDE3.
2. CellDe es capaz de mejorar las soluciones obtenidas NSGA-II, SPEA2, GDE3 y MOCell de acuerdo a las pruebas realizadas, caracterizadas por:
   (a) Benchmark compuesto de problemas con tres objetivos.
   (b) Indicadores de calidad: Epsilon (I_ϵ) e Hypervolume (HV).

RELACIÓN CON ENTREGABLES:

Málaga, November 22^{nd}, 2008

Executive Summary

TITLE: MULTIOBJ-1.2: A Cellular Hybrid Algorithm based on Differential Evolution (CellDE).

ABSTRACT: This document is aimed at describing CellDE, a new cellular hybrid multi-objective algorithm based on the differential evolution mechanism. Furthermore, in order to assessing the performance of CellDE, it is compared against four multi-objective state-of-the-art algorithms: NSGA-II, SPEA2, MOCell, and GDE3. This comparison has been made on the basis of two quality indicators (Epsilon (I_ϵ) e Hypervolume (HV)), and a benchmark composed of 16 problems (DTLZ, and WFG) configured with three objectives.

GOALS:

1. To design a new hybrid cellular algorithm based on the differential evolution mechanism.
2. To solve problems with more than two objective functions.
3. To compare the quality of the obtained results with those obtained by four state-of-the-art algorithms.

CONCLUSIONS:

1. A new algorithm based on the hybridization of MOCell and GDE3 has been designed.
2. CellDE outperforms NSGA-II, SPEA2, GDE3, and MOCell in the following context:
   (a) Benchmark composed of three-objective problems.
   (b) Quality indicators: Epsilon (I_ϵ) e Hypervolume (HV).

RELATION WITH DELIVERABLES:

MULTIOBJ-1.2: A Cellular Hybrid Algorithm based on Differential Evolution (CellDE)

DIRICOM

December 2008

1 Indtroduction

Multi-objective optimization refers to optimizing problems whose formulation involves two or more objectives, which are known as multi-objective optimization problems (MOPs). The solution to these kinds of problem is not usually a single one; instead, a set of *nondominated solutions* has to be found. Each solution in this set is said to be a *Pareto optimum*, and when they are plotted in the objective space they are collectively known as the *Pareto front*.

In the last few years evolutionary algorithms (EAs) have become very popular tools for solving MOPs since they are capable of obtaining the Pareto front in a single run. As a consequence, many multi-objective EAs have appeared in recent years, and the most well-known metaheuristics, such as NSGA-II [1], SPEA2 [2], PAES [3], and many others [4][5], belong to this family of techniques. Most of these algorithms are genetic algorithms (GA), a subclass into EAs.

Our starting point is MOCell [6], a multi-objective cellular GA that is characterized by the use of an external archive to store the non-dominated solutions found during the search and a feedback mechanism in which solutions from this archive randomly replaces existing individuals in the population after each iteration. In order to manage the insertion of solutions in the archive with the goal of obtaining a diverse set, MOCell includes a density estimator based on the crowding distance of NSGA-II [1]. This measure is also used to remove solutions from the archive when it becomes full. MOCell has proven to be very effective in solving bi-objective MOPs; in particular, it provides Pareto fronts with a remarkable uniformity (spread) of their solutions. However, preliminary experiments have revealed that it has difficulties when dealing with three-objective MOPs (namely, those belonging to the DTLZ problem family [7]).

In our research activity, we paid attention to differential evolution (DE) algorithms [8], another kind of EA. DE has been successfully applied as a single-objective optimizer in continuous search problems within the last few years [9], and there are proposals which adapt it to multi-objective optimization [10, 11, 12]. In particular, we focused on the Generalized Differential Evolution 3 (GDE3) algorithm [11]. Preliminary experiments with GDE3 showed that it was able to reach solution sets which are very close to the Pareto front when solving some DTLZ problems.

This work is aimed at designing a metaheuristic capable of producing the same satisfactory results in three-objective MOPs as MOCell achieves in bi-objective problems. Our proposal is a new hybrid metaheuristic, called CellDE, which tries to combine the advantages of both MOCell (good diversity in bi-objective MOPs) and GDE3 (good convergence in three-objective MOPs). The idea is to use MOCell as search engine and hybridizing it with DE, by replacing the typical genetic operators of crossover and mutation of GAs by the reproductive mechanism used in DE. This result summarize the work presented in [19].

2 CellDE

The pseudocode of the algorithm is shown in Algorithm 1. The basic behavior of CellDE is that of a cGA following an asynchronous behavior, in the sense that all the cells are explored sequentially (in the case of synchronous cGAs, the cells are explored in parallel). The MOCell version taken as starting point is based on aMOCell3 [6], which is characterized by using an external archive to store the non-dominated solutions found so far during the search and a feedback mechanism. The aMOCell3 algorithm was originally engineered using the crowding distance as density estimator to manage the diversity in the approximated Pareto front. As it has been reported in the literature [14], this estimator does not perform well with MOPs having more than two objectives. This leads us to use the density estimator of SPEA2 [2] in our approach and also in the aMOCell3 algorithm used in our experiments.

The main difference between CellDE and MOCell (we will refer aMOCell3 as MOCell in the rest of the paper) arises in the creation of new individuals. Instead of using the classical GA operators to generate new individuals, CellDE takes the operators used in DE (as shown in Algorithm 2): three different individuals (the parents) are chosen and the new offspring solution is obtained based on the differences between them.

CellDE starts by creating a population of random solutions and an empty Pareto front (lines 2 and 3 in Algorithm 1). Individuals are arranged in a 2-dimensional grid, defining neighborhood structures over the population.

Algorithm 1 Pseudocode of CellDE.

```
1: proc Steps_Up(CellDE)                    //Algorithm parameters in 'CellDE'
2: population ← randomPopulation()          //Creates a random initial population
3: archive ← createFront()                  //Creates an empty Pareto front
4: while !terminationCondition() do
5:    for individual ← 1 to CellDE.populationSize do
6:       neighborhood←getNeighbors(population,position(individual));
7:       parent1←selection(neighborhood);
8:       parent2←selection(neighborhood);
9:       // parent1 and parent2 may be different
10:      while parent1=parent2 do
11:         parent2←selection(neighborhood);
12:      end while
13:      offspring←differentialEvolution(individual, parent1, parent2);
14:      evaluateFitness(offspring);
15:      insert(position(individual),offspring,population);
16:      addArchive(individual);
17:   end for
18:   population←replaceIndividuals(population,archive);
19: end while
20: end_proc Steps_Up;
```

Algorithm 2 Pseudocode of generating a new solution in DE.

```
r_1, r_2, r_3 ∈ {1, 2, ..., N}, (randomly selected, except mutually different and different from i)
jrand =floor(rand_i[0, 1) · D) + 1
for (j = 1; j ⩽ D; j = j + 1) do
   if (rand_j[0, 1) < CR ∨ j = j_rand) then
      u_{j,i,G} = x_{j,r3,G} + F · (x_{j,r1,G} − x_{j,r2,G})
   else
      u_{j,i,G} = x_{j,i,G}
   end if
end for
```

For each individual $\vec{x}_{i,G}$, two different solutions of the neighborhood are selected (lines 7 and 8) which, along with the current individual, are used as the three parents to create the new offspring (line 13). This is a different approach to the one used in DE, where the three parents exclude the current solution; we take this scheme since it allows to enhance the intensification of the algorithm. The newly generated offspring is evaluated (line 14) and then it replaces the original solution if dominates it, or, if both are non-dominated, it replaces the worst individual in the neighborhood (line 15). After that, the new individual is sent to the archive, where it is checked for its insertion (line 16). Finally, after each generation, a feedback procedure is performed to replace a number of randomly chosen individuals by a number of solutions taken from the archive (line 18).

3 Experimentation

We have chosen several test problems taken from the specialized literature, and, in order to assess how competitive CellDE is, we have compared it to the two reference algorithms in the field, namely NSGA-II and SPEA2, as well as to the base algorithms used for designing CellDE, GDE3 and MOCell. All the algorithms have been implemented in Java using the jMetal framework [15].

The test problems we have used are the three-objective formulations of the Deb-Thiele-Laumanns-Zitzler (DTLZ) benchmark [7] and the Walking-Fish-Group (WFG) problems [13]. A total number of sixteen MOPs has been used to evaluate the five metaheuristics. For assessing the performance of the algorithms, we have used two Pareto-compliant indicators: hypervolume (HV) [16] and additive epsilon indicator ($I^1_{\epsilon+}$) [17]. The latter is an indicator measuring the convergence of the resulting Pareto fronts, while the former measures both convergence and diversity.

We have made 100 independent runs of each experiment, and we have obtained the median, $\tilde{x}$, and interquartile range, IQR, as measures of location (or central tendency) and statistical dispersion, respectively. Since we are dealing with stochastic algorithms and we want to provide the results with confidence, the following statistical analysis has been performed in all this work [18]. Firstly, a Kolmogorov-Smirnov test is applied in order to check whether the values of the results follow a normal (gaussian) distribution or not. If the distribution is normal, the Levene test checks for the homogeneity of the variances. If samples have equal variance (positive Levene test), an ANOVA test is done; otherwise a Welch test is performed. For non-gaussian distributions, the non-parametric Kruskal-Wallis test is used to compare the medians of the algorithms. We always consider a confidence level of 95% (i.e., significance level of 5% or p-value under 0.05) in the statistical tests. Successful tests are marked with '+' symbols in the last column in all the tables containing the results; conversely, '-' means that no statistical confidence was found (p-value > 0.05). The best result for each problem has a gray colored background. For the sake of a better understanding of the results, we have also used a clearer grey background to indicate the second best result.

Table 1: Median and interquartile range of the (additive) Epsilon (I_ϵ) indicator

Problem	NSGA-II $\tilde{x}_{IQR}$	SPEA2 $\tilde{x}_{IQR}$	GDE3 $\tilde{x}_{IQR}$	MOCell $\tilde{x}_{IQR}$	CellDE $\tilde{x}_{IQR}$	
DTLZ1	7.62e-2 $_{7.2e-2}$	4.16e-2 $_{8.4e-3}$	4.80e-2 $_{6.6e-3}$	5.35e-1 $_{5.1e-1}$	3.34e-2 $_{3.3e-3}$	+
DTLZ2	1.24e-1 $_{2.0e-2}$	8.20e-2 $_{9.5e-3}$	1.17e-1 $_{1.7e-2}$	7.99e-2 $_{8.5e-3}$	7.62e-2 $_{8.8e-3}$	+
DTLZ3	4.51e+0 $_{2.7e+0}$	4.73e+0 $_{3.0e+0}$	1.36e+1 $_{5.2e+0}$	1.67e+1 $_{7.8e+0}$	3.55e+0 $_{3.3e+0}$	+
DTLZ4	1.12e-1 $_{2.4e-2}$	7.93e-2 $_{5.6e-1}$	1.08e-1 $_{1.9e-2}$	6.92e-2 $_{1.0e-2}$	6.77e-2 $_{8.6e-3}$	+
DTLZ5	1.07e-2 $_{2.6e-3}$	7.74e-3 $_{1.5e-3}$	5.58e-3 $_{4.8e-4}$	8.08e-3 $_{1.6e-3}$	6.55e-3 $_{1.1e-3}$	+
DTLZ6	8.57e-1 $_{1.3e-1}$	7.82e-1 $_{6.3e-2}$	5.10e-3 $_{5.5e-4}$	1.72e+0 $_{1.5e-1}$	6.00e-3 $_{7.9e-4}$	+
DTLZ7	1.27e-1 $_{4.5e-2}$	9.82e-2 $_{1.2e-2}$	1.20e-1 $_{3.6e-2}$	1.15e-1 $_{3.0e-2}$	8.42e-2 $_{1.6e-2}$	+
WFG1	5.66e-1 $_{6.8e-2}$	6.56e-1 $_{1.1e-1}$	7.76e-1 $_{1.1e-1}$	6.30e-1 $_{1.8e-1}$	1.03e+0 $_{1.5e-1}$	+
WFG2	3.23e-1 $_{6.4e-2}$	2.37e-1 $_{3.4e-2}$	3.02e-1 $_{4.5e-2}$	2.56e-1 $_{3.8e-2}$	2.52e-1 $_{3.9e-2}$	+
WFG3	1.24e-1 $_{3.5e-2}$	9.22e-2 $_{1.7e-2}$	1.08e-1 $_{3.6e-2}$	8.57e-2 $_{1.8e-2}$	1.04e-1 $_{3.0e-2}$	+
WFG4	4.32e-1 $_{7.8e-2}$	3.26e-1 $_{3.8e-2}$	4.21e-1 $_{1.0e-1}$	2.95e-1 $_{4.3e-2}$	3.10e-1 $_{4.1e-2}$	+
WFG5	4.71e-1 $_{7.8e-2}$	3.52e-1 $_{4.6e-2}$	4.34e-1 $_{6.4e-2}$	3.44e-1 $_{4.2e-2}$	3.30e-1 $_{4.7e-2}$	+
WFG6	4.31e-1 $_{6.7e-2}$	3.30e-1 $_{4.9e-2}$	3.94e-1 $_{6.2e-2}$	3.13e-1 $_{4.4e-2}$	2.81e-1 $_{3.6e-2}$	+
WFG7	4.65e-1 $_{8.7e-2}$	3.37e-1 $_{3.9e-2}$	4.57e-1 $_{1.1e-1}$	3.07e-1 $_{3.8e-2}$	2.95e-1 $_{3.7e-2}$	+
WFG8	7.51e-1 $_{9.2e-2}$	6.22e-1 $_{1.4e-1}$	7.56e-1 $_{5.4e-2}$	6.26e-1 $_{1.6e-1}$	6.38e-1 $_{3.3e-2}$	+
WFG9	4.39e-1 $_{7.2e-2}$	3.28e-1 $_{4.2e-2}$	4.25e-1 $_{5.8e-2}$	3.13e-1 $_{4.5e-2}$	3.14e-1 $_{3.7e-2}$	+

Table 2: Median and interquartile range of the HV indicator.

Problem	NSGA-II $\tilde{x}_{IQR}$	SPEA2 $\tilde{x}_{IQR}$	GDE3 $\tilde{x}_{IQR}$	MOCell $\tilde{x}_{IQR}$	CellDE $\tilde{x}_{IQR}$	
DTLZ1	7.22e-1 $_{1.0e-1}$	7.69e-1 $_{1.5e-2}$	7.62e-1 $_{6.0e-3}$	0.00e+0 $_{1.0e-1}$	7.86e-1 $_{7.9e-4}$	+
DTLZ2	3.73e-1 $_{8.3e-3}$	4.05e-1 $_{2.6e-3}$	3.74e-1 $_{6.3e-3}$	4.10e-1 $_{2.0e-3}$	4.16e-1 $_{1.3e-3}$	+
DTLZ3	–	–	–	–	–	-
DTLZ4	3.74e-1 $_{7.6e-3}$	3.98e-1 $_{1.9e-1}$	3.71e-1 $_{5.9e-3}$	4.05e-1 $_{1.9e-3}$	4.07e-1 $_{1.4e-3}$	+
DTLZ5	9.28e-2 $_{3.0e-4}$	9.32e-2 $_{1.9e-4}$	9.39e-2 $_{7.0e-5}$	9.33e-2 $_{1.7e-4}$	9.36e-2 $_{6.9e-5}$	+
DTLZ6	–	–	9.49e-2 $_{4.8e-5}$	–	9.46e-2 $_{8.1e-5}$	-
DTLZ7	2.80e-1 $_{6.0e-3}$	2.90e-1 $_{3.5e-3}$	2.92e-1 $_{2.8e-3}$	2.81e-1 $_{7.2e-3}$	3.03e-1 $_{2.4e-3}$	+
WFG1	7.71e-1 $_{5.2e-2}$	6.75e-1 $_{7.4e-2}$	6.42e-1 $_{5.4e-2}$	7.17e-1 $_{1.3e-1}$	5.27e-1 $_{1.1e-1}$	+
WFG2	9.01e-1 $_{4.7e-3}$	9.13e-1 $_{1.9e-3}$	9.05e-1 $_{3.3e-3}$	9.12e-1 $_{1.7e-3}$	9.14e-1 $_{1.8e-3}$	+
WFG3	3.19e-1 $_{2.5e-3}$	3.11e-1 $_{2.7e-3}$	3.23e-1 $_{1.5e-3}$	3.15e-1 $_{1.6e-3}$	3.11e-1 $_{4.6e-3}$	+
WFG4	3.65e-1 $_{8.2e-3}$	3.92e-1 $_{4.8e-3}$	3.52e-1 $_{8.6e-3}$	4.07e-1 $_{2.3e-3}$	3.95e-1 $_{4.4e-3}$	+
WFG5	3.41e-1 $_{9.4e-3}$	3.68e-1 $_{6.9e-3}$	3.55e-1 $_{4.0e-3}$	3.68e-1 $_{4.8e-3}$	3.71e-1 $_{1.9e-3}$	+
WFG6	3.64e-1 $_{1.0e-2}$	3.91e-1 $_{1.4e-2}$	3.81e-1 $_{9.0e-3}$	3.97e-1 $_{1.5e-2}$	4.16e-1 $_{2.6e-3}$	+
WFG7	3.58e-1 $_{1.0e-2}$	3.83e-1 $_{5.5e-3}$	3.63e-1 $_{7.8e-3}$	4.00e-1 $_{3.2e-3}$	4.07e-1 $_{2.5e-3}$	+
WFG8	2.42e-1 $_{6.7e-3}$	2.69e-1 $_{9.2e-3}$	2.40e-1 $_{5.5e-3}$	2.69e-1 $_{7.7e-3}$	2.59e-1 $_{5.1e-3}$	+
WFG9	3.57e-1 $_{7.2e-3}$	3.77e-1 $_{3.8e-3}$	3.61e-1 $_{5.4e-3}$	3.86e-1 $_{5.9e-3}$	3.86e-1 $_{2.7e-3}$	+

4 Computational Results

This section is devoted to the evaluation of CellDE. We start by analyzing the Epsilon (I_ϵ) quality indicator and, after that, we pay attention to the results obtained with the hypervolume (HV).

The results of the I_ϵ indicator are included in Table 1. We observe that CellDE obtains the best (lowest) values in eight out of the sixteen problems evaluated and the second best results in five cases. MOCell is the second best algorithm (three best results and seven second best values) followed by SPEA2 (best value in two out of the sixteen problems evaluated and the second best value in three other problems). GDE only yields the best values in two problems. NSGA-II is the technique providing the poorest fronts, which confirms the fact that this algorithm has difficulties when solving MOPs having more than two objectives.

We analyze now the results obtained after applying the HV indicator (see Table 2). It can be seen that CellDE clearly outperforms the other algorithms, obtaining the best (highest) values in nine out of the sixteen MOPs evaluated, yielding also the second best values in three other problems. MOCell can be considered as the second most competitive algorithm according to HV since, although it reaches the best HV value in only a single MOP, it is the second best in eight out of the sixteen problems. GDE3 gets the best value in three MOPs, and the second best value only in one case, while SPEA2 obtains the best value in only one problem and the second best value in two cases. The least algorithm with respect to this indicator is NSGA-II, which only reaches the best value in one problem, yielding also the second best value in another one. As to GDE3 and MOCell, the base algorithms for CellDE, we can state that the search capabilities of the new approach improves significantly those of the two former ones according to HV. We explain now the meaning of the '−' symbol in Table 2. Since the HV indicator is not free from the arbitrary scaling of the objectives, the resulting Pareto fronts of the algorithms have to be normalized. In this normalization process, the nondominated solutions that are outside the limits of the true Pareto front are not considered to compute the HV value because, otherwise, the obtained values would be unreliable.

To illustrate the working principles of CellDE, we include in Fig. 1 the Pareto fronts produced by the different algorithms evaluated when solving problem DTLZ1. We observe that the fronts obtained by CellDE and SPEA2 have a better distribution of solutions than the other ones. Furthermore, in the case of CellDE, all the solutions have converged towards the true Pareto front, while in the SPEA2 front some solutions have not.

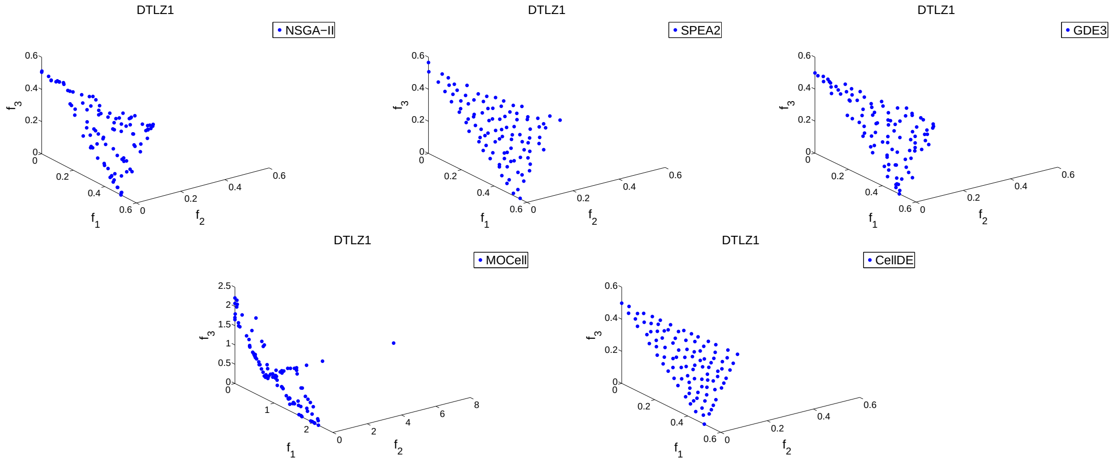

Figure 1: Front obtained when solving DTLZ1. From left to right, from top to bottom: NSGA-II, SPEA2, GDE3, MOCell, CellDE

5 Conclusions

In this work we have proposed a new algorithm called CellDE, which hybridizes the behavior of a cellular GA with a DE algorithm. It has been evaluated using a benchmark composed of sixteen three-objective optimization problems.

To assess how competitive CellDE is, we have compared it to four state-of-the-art algorithms, NSGA-II, SPEA2, MOCell, and GDE3 and MOCell, being the last two ones the starting point to design our algorithm. The obtained results show that CellDE clearly outperforms the other techniques according to the parameter settings, problems, and quality indicators used.

References

[1] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2) (2002) 182–197

[2] Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), ETH Zurich (2001)

[3] Knowles, J., Corne, D.: The pareto archived evolution strategy: A new baseline algorithm for multiobjective optimization. In: Proceedings of the 1999 Congress on Evolutionary Computation, Piscataway, NJ, IEEE Press (1999) 9–105

[4] Coello, C., Van Veldhuizen, D., Lamont, G.: Evolutionary Algorithms for Solving Multi-Objective Problems. 2nd Edition. Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers (2007)

[5] Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons (2001)

[6] Nebro, A.J., Durillo, J.J., Luna, F., Dorronsoro, B., Alba, E.: Design issues in a multiobjective cellular genetic algorithm. In: Evolutionary Multi-Criterion Optimization (EMO 2007). LNCS 4403 (2007) 126 – 140

[7] Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Test Problems for Evolutionary Multi-Objective Optimization. Technical Report 112, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland (2001)

[8] Storn, R., Price, K.: Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, Berkeley, CA (1995)

[9] Price, K., Storn, R., Lampinen, J.: Differential Evolution A Practical Approach to Global Optimization. Natural Computing Series. Springer-Verlag, Berlin, Germany (2005)

[10] Lampinen, J.: De's selection rule for multiobjective optimization. Technical report, Lappeenranta University of Technology, Department of Information Technology (2001)

[11] Kukkonen, S., Lampinen, J.: GDE3: The third Evolution Step of Generalized Differential Evolution. In: IEEE Congress on Evolutionary Computation (CEC'2005). (2005) 443 – 450

[12] Hernández-Díaz, A.G., Santana-Quintero, L.V., Coello, C.A.C., Caballero, R., Molina, J.: A new proposal for multi-objective optimization using differential evolution and rough sets theory. In: Conference on Genetic and Evolutionary Computation (GECCO'06). (2006) 675 – 682

[13] Huband, S., Hingston, P., Barone, L., While, L.: A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Transactions on Evolutionary Computation **10**(5) (October 2006) 477–506

[14] Kukkonen, S., Deb, K.: Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. In: 2006 IEEE Congress on Evolutionary Computation (CEC'2006). (2006) 1179 – 1186

[15] Durillo, J.J., Nebro, A.J., F.Luna, Dorronsoro, B., Alba, E.: jMetal: a java framework for developing multi-objective optimization metaheuristics. Technical Report ITI-2006-10, Dpto. de Lenguajes y Ciencias de la Computación, University of Málaga (2006)

[16] Zitzler, E., Thiele, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. IEEE Transactions on Evolutionary Computation **3**(4) (1999) 257–271

[17] Knowles, J., Thiele, L., Zitzler, E.: A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. Technical Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich (2006)

[18] Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7** (2006) 1–30

[19] Durillo, J. J., Nebro, A.J., Luna, F., Alba, E. Solving Three-Objective Optimization Problems Using a new Hybrid Cellular Genetic Algorithm. PPSN X. LNCS, Vol. Volume 5199/2008, pp: 661-670. Dortmund, September 2008.