

pMODE-LD+SS: An Effective and Efficient Parallel Differential Evolution Algorithm for Multi-Objective Optimization

Alfredo Arias Montaño^{1*}, Carlos A. Coello Coello^{1**}, and Efrén Mezura-Montes^{2***}

¹ CINVESTAV-IPN, Av. IPN 2508, San Pedro Zacatenco, Gustavo A. Madero, México D.F. 07360, MEXICO

aarias@computacion.cs.cinvestav.mx, ccoello@cs.cinvestav.mx

² LANIA A.C. Rébsamen 80, Centro, Xalapa, Veracruz, 91000, MEXICO
emezura@lania.mx

Abstract. This paper introduces a novel Parallel Multi-Objective Evolutionary Algorithm (pMOEA) which is based on the island model. The serial algorithm on which this approach is based uses the differential evolution operators as its search engine, and includes two mechanisms for improving its convergence properties (through local dominance and environmental selection based on scalar functions). Two different parallel approaches are presented. The first aims at improving effectiveness (i.e., for better approximating the Pareto front) while the second aims to provide a better efficiency (i.e., by reducing the execution time through the use of small population sizes in each sub-population). To assess the performance of the proposed algorithms, we adopt a set of standard test functions and performance measures taken from the specialized literature. Results are compared with respect to its serial counterpart and with respect to three algorithms representative of the state-of-the-art in the area: NSGA-II, MOEA/D and MOEA/D-DE.

1 Introduction

Multi-objective evolutionary algorithms (MOEAs) have been found to be very suitable for solving a wide variety of engineering optimization problems, because of their generality, their ease of use and their relatively low susceptibility to the specific features of the search space of the problem to be solved [1]. Nonetheless, they are normally computationally expensive due to several reasons: (1) real-world optimization problems typically involve high-dimensional search spaces and/or a large number of objective functions, (2) they require finding a set of solutions instead of only one, often requiring, in consequence, large population

* The first author acknowledges support from CONACyT and IPN to pursue graduate studies in computer science at CINVESTAV-IPN.

** The second author acknowledges support from CONACyT project no. 103570.

*** The third author acknowledges support from CONACyT project No. 79809.

sizes, and (3) frequently, the task of evaluating the objective functions demands high computational costs (e.g., complex computer simulations are required). All these factors decrease the utility of serial MOEAs for its use in real-world engineering Multi-objective Optimization Problems (MOPs). In order to reduce the execution time required to solve these problems two main types of approaches have been normally adopted³: (1) Enhance the MOEA’s design, namely improving its convergence properties, so that the number of objective function evaluations can be reduced, and, (2) Use of parallel programming techniques, i.e., to adopt a parallel or distributed MOEA.

Based on the above, the major aim of the present work is to develop two different schemes for improving the performance of a pMOEA. The first is designed for improving effectiveness (i.e., for better approximating the Pareto front), while the second is designed for improving efficiency (i.e., for reducing the execution time by using small population sizes in each sub-population). Either approach (or both) can be of interest in solving real-world engineering MOPs. The two proposed schemes are based on the *island paradigm*, and use the multi-objective differential evolution algorithm MODE-LD+SS [2] as its search engine. The proposed schemes are evaluated using standard test functions and performance measures taken from the specialized literature. The results obtained by the two proposed pMOEAs are compared with respect to their serial counterpart and with respect to the NSGA-II [3], MOEA/D [4], and MOEA/D-DE [5].

The remainder of the paper is organized as follows. In Section 2, the most relevant previous related work on island-based pMOEAs is presented. Section 3 is devoted to describe the proposed approach. Then, the experimental setup is presented in Section 4. In Section 5 the obtained results are presented and discussed. Finally, in Section 6 our conclusions and the corresponding future work is highlighted.

2 Previous related work

A pMOEA can be useful to solve problems faster, but also for generating novel and more efficient search schemes, i.e., a pMOEA can be more effective than its sequential counterpart, even when executed in a single processor [6]. From the specialized literature, four major pMOEA paradigms are commonly used [7]: (i) “Master-Slave,” (ii) “Island,” (iii) “Diffusion,” and (iv) “hierarchical” or “hybrid”. A comprehensive review of these paradigms can be found in [1, 7]. This paper focuses on the *Island Model*, which is based on the phenomenon of natural populations evolving independently. In each island, a serial MOEA is executed for a predefined number of generations called *epoch*. At the end of an epoch, communication between neighboring islands is allowed. In this communication, individuals (or copies of them in the case of pollination) can migrate from its

³ Our discussion here is focused exclusively on MOEAs that use exact objective function values, since fitness approximation schemes and surrogate models can also be used to deal with expensive MOPs.

actual island to a different one according to a predefined *migration topology* which determines the migration path along which individuals can move.

Kamiura et al. [8] presented a pMOEA called MOGADES (Multi-Objective Genetic Algorithm with Distributed Environment Scheme). In this pMOEA, the population is divided into M islands, and in each of them the MOP is converted into a scalar one, i.e., a different weight vector is assigned to each island. The aim of this algorithm is that each island can capture a different region of the Pareto front. One important aspect in this approach is that when migration occurs, the weights for each island are varied. A major drawback for this approach is that a good distribution of solutions cannot be guaranteed as it depends on the dynamics of the evolutionary system, i.e., of the weight vector variation.

Streichert et al. [9] proposed a pMOEA, which combines an island model with the “divide and conquer” principle. This approach partitions the population using a clustering algorithm (k -means), with the aim of assigning to each island, the search task of a particular Pareto front region. In this approach, at each epoch, the sub-populations are gathered by a master process for performing the clustering/distributing process. The individuals in each island are kept within their assigned Pareto front region using zone constraints. The main drawback of this approach is that *a priori* knowledge of the Pareto front shape is needed to define the zone constraints.

Zahaire and Petcu [10] developed the *multi-population APDE* (APDE stands for Adaptive Pareto Differential Evolution). This approach consists of dividing the main population into sub-populations (islands), each of equal size. In each island, a serial version of the APDE is executed with its own set of randomly initialized adaptive parameters, and is evolved for an epoch. Afterwards, a migration process is started. This process is based on a random connection topology, i.e., each individual from each sub-population can be swapped (with a given migration probability) with a randomly selected individual from another randomly selected island.

3 Our Proposed Approach

The first mechanism is as follows. In our proposed parallel algorithm, each island runs an approach called MODE-LD+SS [2], which adopts operators from differential evolution using the DE/RAND/1/bin scheme. Algorithm 1 shows the basic (serial version) pseudo-code of our proposed MODE-LD+SS approach. In Algorithm 1, the solution vectors $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$ are selected from the current population, only if they are locally nondominated in their neighborhood \aleph . Local dominance is defined as follows:

Definition 1. Pareto Local Dominance Let \mathbf{x} be a feasible solution, $\aleph(\mathbf{x})$ be a neighborhood structure for \mathbf{x} in the decision space, and $\mathbf{f}(\mathbf{x})$ a vector of objective functions.

- We say that a solution \mathbf{x} is locally nondominated with respect to $\aleph(\mathbf{x})$ if and only if there is no \mathbf{x}' in the neighborhood of \mathbf{x} such that $\mathbf{f}(\mathbf{x}') \prec \mathbf{f}(\mathbf{x})$

The neighborhood structure is defined as the NB closest individuals to a particular solution. Closeness is measured using the Euclidean distance between solutions. The major aim of using the local dominance concept, as defined above, is to exploit good individuals' genetic information in creating DE trial vectors, and the associated offspring, which might help to improve the MOEA convergence rate toward the Pareto front. From Algorithm 1, it can be noted that this mechanism has a stronger effect during the earlier generations, where the portion of nondominated individuals is low in the global population, and progressively weakens, as the number of nondominated individuals grows during the evolutionary process. This mechanism is automatically switched off, once all the individuals in the population become nondominated, and has the possibility to be switched on, as some individuals become dominated. Additionally, the diversity of the created offspring can be controlled by the local dominance neighborhood size NB . Low values of NB will increase the diversity of the offspring, and viceversa.

The second mechanism that is introduced in MODE-LD+SS is called *environmental selection based on a scalar function*, and is based on the Tchebycheff scalarization function given by [4]:

$$g(\mathbf{x}|\lambda^j, z^*) = \max_{1 \leq i \leq m} \{\lambda_i^j |f_i(x) - z_i^*|\} \quad (1)$$

In the above equation, $\lambda^j, j = 1, \dots, N$ represents the weight vectors used to distribute the solutions along the whole Pareto front (see Figure 1(a)). z^* corresponds to a reference point, defined in objective function space and determined with the minimum objective values of the population. This reference point is updated at each generation, as the evolution progresses. The procedure *MinimumTchebycheff*(Q, λ^j, z^*) finds, from the set Q (the combined population consisting on the actual parents and the created offspring), the solution vector that minimizes equation (1) for each weight vector λ^j and the reference point z^* .

Based on the serial MOEA previously described, we present here two parallelization schemes. The first is designed for improving effectiveness and is called pMODE-LD+SS(A). The second is designed for improving efficiency, and is called pMODE-LD+SS(B). Both of them share the following characteristics:

- Use of a “random pair-islands” bidirectional migration scheme. In this scheme, at each epoch, pairs of islands are randomly selected. Then, the communication is performed between each pair of islands. Migrants from one island are considered as immigrants in the receptor island, and viceversa.
- Use of a pollination scheme, i.e., copies of selected migrants are sent, while the original individuals are retained in their own population.
- The migration policy is based on randomly selected individuals.
- The replacement policy is based on the environmental selection mechanism adopted in the serial version running in each island. In this case, immigrants are added to the receptor island's population, and the environmental selection process is applied to this extended population.

Algorithm 1 MODE-LD+SS

```

1: INPUT:
    $N$  = Population Size
    $F$  = Scaling factor
    $CR$  = Crossover Rate
    $\lambda[1, \dots, N]$  = Weight vectors
    $NB$  = Neighborhood Size
    $GMAX$  = Maximum number of generations
2: OUTPUT:
    $PF$  = Pareto front approximation
3: Begin
4:  $g \leftarrow 0$ 
5: Randomly create  $P_i^g, i = 1, \dots, N$ 
6: Evaluate  $P_i^g, i = 1, \dots, N$ 
7: while  $g < GMAX$  do
8:    $\{LND\} = \{\emptyset\}$ 
9:   for  $i = 1$  to  $N$  do
10:     $DetermineLocalDominance(P_i^g, NB)$ 
11:    if  $P_i^g$  is locally nondominated then
12:       $\{LND\} \leftarrow \{LND\} \cup P_i^g$ 
13:    end if
14:  end for
15:  for  $i = 1$  to  $N$  do
16:    Randomly select  $u_1, u_2$ , and  $u_3$  from  $\{LND\}$ 
17:     $v \leftarrow CreateMutantVector(u_1, u_2, u_3)$ 
18:     $P_i^{g+1} \leftarrow Crossover(P_i^g, v)$ 
19:    Evaluate  $P_i^{g+1}$ 
20:  end for
21:   $Q \leftarrow P^g \cup P^{g+1}$ 
22:  Determine  $z^*$  for  $Q$ 
23:  for  $i = 1$  to  $N$  do
24:     $P_i^{g+1} \leftarrow MinimumTchebycheff(Q, \lambda^i, z^*)$ 
25:     $Q \leftarrow Q \setminus P_i^{g+1}$ 
26:  end for
27:   $PF \leftarrow Q$ 
28: end while
29: Return  $PF$ 
30: End

```

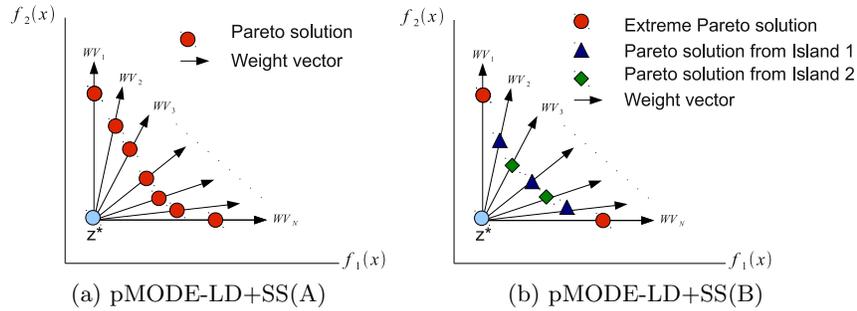


Fig. 1. Weight vectors distribution

The main difference between the two proposed approaches is on the weight vectors distribution used. The pMODE-LD+SS(A) approach can be seen as the serial version of MODE-LD+SS running in p processors and exchanging information among them. For this approach, the same weight vector distribution (see Figure 1(a)) is used in each island. For maintaining diversity of the global population and to evolve each island in an independent manner, different seed values are used in the islands' random numbers generators. In the second case, for the pMODE-LD+SS(B) approach, each island is also instructed to search for the whole Pareto front, but in this case, using a reduced population and different weight vectors sets. It is important to note that all islands contain weight vectors for searching the extreme Pareto solutions. The main idea for the second parallel approach is that the combination of all islands' weight vectors covers the whole Pareto front region. Figure 1(b) illustrates this situation for the case of a bi-objective MOP with two islands participating in the pMOEA.

4 Experimental setup

In order to validate the two proposed parallel approaches, their results are compared with respect to those generated by their serial counterpart (MODE-LD+SS), and to NSGA-II [3], MOEA/D [4], and MOEA/D-DE [5] which are MOEAs representative of the state-of-the-art in multiobjective evolutionary optimization. Our approaches were validated using nine test problems: five from the ZDT (Zitzler-Deb-Thiele) set with 2 objectives (ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6⁴), and four more from the DTLZ (Deb-Thiele-Laumanns-Zitzler) set with 3 objectives (DTLZ1, DTLZ2, DTLZ3, and DTLZ4). The selected test functions comprise different difficulties such as convex, concave, and disconnected Pareto fronts, as well as problems with multiple fronts. Two performance measures were adopted in order to assess our results: *Hypervolume (Hv)* and *Two Set Coverage (C-Metric)*. A description of these performance measures are omitted here but can be found elsewhere [1]. The Hv measure uses a reference point in the objective space which was set to (1.05,1.05) for all the 2-objective MOPs, and to (5.0,5.0,5.0) in all the 3-objective MOPs. In the case of the C-Metric, the A reference set is considered as the true Pareto front which is known for all the MOPs used in the experiments. Thus, the C-Metric can be considered as a measure for the ability of the algorithm to find solutions that are nondominated with respect to the Pareto optimal set (i.e., solutions that also belong to the Pareto optimal set).

5 Results and discussion

In this section we present the results obtained by the proposed parallel approaches. As a first step, the serial version of MODE-LD+SS is compared with

⁴ ZDT5 is a binary problem and was not included because we adopted real-numbers encoding in our experiments.

respect to NSGA-II, MOEA/D and MOEA/D-DE. Then, the number of islands, epoch and migration rate adopted in the parallel approaches are tuned by means of an empirical study, using ZDT1. Finally, the results obtained with the two parallel approaches are presented and compared to those of the serial version of MODE-LD+SS, and those obtained by NSGA-II, MOEA/D and MOEA/D-DE. These comparisons are based on the average results from 32 independent runs executed by each algorithm and for each MOP.

Comparison of serial versions

Table 1 presents the results of the serial versions for NSGA-II, MOEA/D, MOEA/D-DE and MODE-LD+SS. The population size in all the algorithms was set to $N=50$ for all the 2-objective MOPs, and 153 for all the 3-objective MOPs the maximum number of generations was set to $GMAX = 150$ for all problems, except for ZDT4 and DTLZ3, where we used $GMAX = 300$. The common parameters for NSGA-II, MOEA/D and MOEA/D-DE were: crossover probability $p_c = 1.0$; mutation probability $p_m = 1/NVARS$ (NVARS correspond to the number of decision variables for each MOP); distribution index for crossover $\eta_c = 15$; and distribution index for mutation $\eta_m = 20$. As for the MOEA/D and MOEA/D-DE the replacing neighborhood size was set as indicated in [4] and [5], respectively. For the MODE-LD+SS algorithm, we used: $F = 0.5$ for all MOPs; $CR = 0.5$ for all MOPs except for ZDT4, where $CR = 0.3$ was used; Neighborhood size $NB = 5$ for all MOPs except for ZDT4, where we used $NB = 1$. From the results presented in Table 1, it can be observed that MODE-LD+SS obtains the best results in 6 of 9 MOPs for the Hv measure. It also obtains the best results in 8 of 9 MOPs regarding the C-Metric, which indicates that it converged closer to the true Pareto front.

Parameters for the pMOEA approaches

For any pMOEA approach based on the island model, additional to the parameters required by its serial counterpart, we have to define the number of islands, migration rate, and epoch period. The choice of these parameters has a great influence in the performance of the pMOEA and is problem dependent. For selecting a set of parameters to be used in the present work, ZDT1 was selected to conduct an experimental study for assessing how the parameters affected performance with respect to the serial version. For this study, the following set of parameters was used: epochs = 10, 20, and 50 generations; migration rate $MR = 0.1, 0.2, 0.3$ and 0.5 ; and number of islands $NI = 4, 6,$ and 8 . All the combinations were tested. The parameters for population size, maximum number of generations, F , CR , and NB were set the same for all the islands, as in the serial version previously described. From the results of this study, and regarding the C-Metric, it was observed that high migration rates with shorter epoch periods produce the best improvements with respect to the serial version. However, this can lead to higher communication costs. From the study, the final set of parameters selected

Table 1. Comparison of Hv and C-Metric measures for the serial versions

Hypervolume measure								
Function	NSGA-II		MOEA-D		MOEA-D-DE		MODE-LD+SS	
	Mean	σ	Mean	σ	Mean	σ	Mean	σ
ZDT1	0.740382	0.003323	0.716729	0.024506	0.583847	0.076507	0.757395	0.000397
ZDT2	0.377348	0.070194	0.176615	0.079320	0.082341	0.115367	0.424895	0.000331
ZDT3	0.604214	0.003199	0.585094	0.023488	0.277813	0.111381	0.613846	0.000307
ZDT4	0.073098	0.122631	0.730980	0.016966	0.450990	0.215977	0.349325	0.285549
ZDT6	0.292164	0.020894	0.375312	0.007755	0.239793	0.084688	0.407638	0.000009
DTLZ1	124.139600	1.113898	124.969600	0.000768	119.402900	7.771898	124.967700	0.000383
DTLZ2	123.972600	0.124088	124.397400	0.001778	124.353700	0.027743	124.397600	0.003356
DTLZ3	80.131930	39.091680	124.338100	0.250190	85.976200	54.287600	124.396900	0.003004
DTLZ4	123.934300	0.125475	124.400100	0.002818	124.387900	0.003452	124.393900	0.002684

C-Metric measure								
Function	NSGA-II		MOEA-D		MOEA-D-DE		MODE-LD+SS	
	Mean	σ	Mean	σ	Mean	σ	Mean	σ
ZDT1	0.994591	0.008785	0.997234	0.007463	1.000000	0.000000	0.748125	0.153569
ZDT2	1.000000	0.000000	0.208557	0.140626	1.000000	0.000000	0.586492	0.100261
ZDT3	0.931490	0.047844	0.813861	0.121330	1.000000	0.000000	0.384729	0.092223
ZDT4	1.000000	0.000000	0.975157	0.088624	1.000000	0.000000	0.845625	0.364496
ZDT6	0.975723	0.008476	0.978242	0.001639	0.989831	0.016529	0.000625	0.003536
DTLZ1	0.535550	0.134412	0.340389	0.234715	0.807088	0.113710	0.021434	0.014189
DTLZ2	0.447368	0.035370	0.211798	0.040208	0.678562	0.053395	0.171215	0.009018
DTLZ3	1.000000	0.000000	0.725727	0.179974	0.972366	0.063160	0.160711	0.007169
DTLZ4	0.453536	0.058747	0.205555	0.036333	0.554462	0.051949	0.156578	0.008628

correspond to the following: Number of Islands = 6; epoch = 10 generations, and migration rate = 0.4. This will be used in assessing the two parallel approaches proposed here. Table 2 shows the results of the two proposed parallel approaches.

pMOEA for effectiveness improvement

From Table 2, it can be observed that the approach designed for effectiveness improvement produced better Hv values in 4 of the 9 MOPs (ZDT1, ZDT3, ZDT4, and ZDT6), while improving the C-Metric in 7 of the 9 MOPs (ZDT1, ZDT4, ZDT6, DTLZ1, DTLZ2, DTLZ3, DTLZ4), with respect to the serial version of MODE-LD+SS. One important result to remark from this parallel approach, is its ability to reach the true Pareto front of ZDT4 and ZDT6 in the 32 runs performed, as indicated by the mean and standard deviations for the C-Metric for these two MOPs.

pMOEA for efficiency improvement

For this approach, each island uses a reduced population size of $N = 10$ for the 2-objective MOPs and of $N = 28$ for the 3-objective MOPs. Since we used 6 islands, the global population consists of 60 individuals for the 2-objective MOPs, and of 168 individuals for the 3-objective MOPs. Considering that the global population size grows, the maximum number of generations used in pMODE-

Table 2. Results for MODE-LD+SS(A) and pMODE-LD+SS(B)

Function	pMODE-LD+SS(A)				pMODE-LD+SS(B)				
	Hv		C-Metric		Hv		C-Metric		Speed-up
	Mean	σ	Mean	σ	Mean	σ	Mean	σ	
ZDT1	0.757675	0.000115	0.623750	0.105517	0.750276	0.002323	0.983219	0.043928	2.9977
ZDT2	0.424363	0.000310	0.766844	0.105424	0.421071	0.001954	0.815243	0.099875	2.6918
ZDT3	0.614156	0.000201	0.385278	0.078578	0.608588	0.001898	0.741921	0.122588	2.5434
ZDT4	0.758770	0.000006	0.000000	0.000000	0.034668	0.119046	1.000000	0.000000	2.3341
ZDT6	0.407650	0.000001	0.000000	0.000000	0.406966	0.000241	0.002552	0.006860	2.5110
DTLZ1	124.967400	0.000238	0.011863	0.004840	124.970200	0.000681	0.022575	0.009888	4.9934
DTLZ2	124.391900	0.002976	0.162893	0.008310	124.404200	0.001968	0.175189	0.008008	4.7269
DTLZ3	124.389000	0.003113	0.155273	0.005601	124.015800	1.228651	0.242665	0.247538	4.8357
DTLZ4	124.389500	0.002848	0.154082	0.008518	124.402700	0.002529	0.149008	0.008031	4.8949

LD+SS(B) were reduced accordingly to obtain an equivalent number of objective function evaluations as in the serial version. However, once the islands' populations are gathered and a global environmental selection is performed, the maximum population size reported for this approach is of 50 solutions for the 2-objective MOPs, and 153 for the 3-objective MOPs. This latter condition is due to the fact that each island searches for the Pareto extreme solutions (there are redundant solutions which are filtered out). The parameters for F, and CR were set the same as in the serial version for all islands. However, due to the reduction in island population size, the parameter NB was set to 1 in all MOPs. In Table 2, the estimated average parallel Speed-Up measure is reported for all MOPs used. Also from this table, it can be seen that the approach designed for efficiency improvement produced better Hv values in 3 of the MOPs adopted (DTLZ1, DTLZ2, and DTLZ4). By taking a closer look to the results for the Hv metric for ZDT1, ZDT2, ZDT3 and ZDT6, it can be seen that pMODE-LD+SS(B) obtained values very close to those of the serial version (MODE-LD+SS), even when each island was using a small population size.

6 Conclusions and future work

We have introduced a new pMOEA, called pMODE-LD+SS. For it, two different parallel schemes were proposed, aiming at improving: (a) effectiveness and (b) efficiency, with respect to its serial version, called MODE-LD+SS. From the results presented in the previous section, the first goal was achieved in 4 and 7 of 9 test problems adopted, when considering the Hv and C-Metric performance measures, respectively. It is worth noting that, in some cases the improvement achieved has been quite significant. Regarding the second goal, and even when each island is using a reduced population size, the second approach is able to obtain better results for the Hv measure in 3 of the 9 test problems adopted. Additionally, in 3 other problems, the values attained are very similar to those obtained by the serial version. From the above, we can conclude that the proposed algorithm has good properties both in terms of effectiveness and efficiency. In the present work, the proposed algorithm was run with parameters derived from empirical tests. However, a thorough statistical analysis is required in order to identify the most appropriate parameters to be adopted, and to relate more

closely such parameter values to specific types of test problems. These tasks will be part of our future work. Given the good convergence properties of the proposed algorithm and its ability to improve both effectiveness and efficiency in the test problems adopted, it is also desirable to test them in real-world optimization problems, and that is actually part of our ongoing research. Finally, we also plan to compare our proposed pMOEA with respect to other pMOEAs currently available in the specialized literature.

References

1. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. Second edn. Springer, New York (2007) ISBN 978-0-387-33254-3.
2. Arias Montaña, A., Coello Coello, C.A., Mezura-Montes, E.: MODE-LD+SS: A Novel Differential Evolution Algorithm Incorporating Local Dominance and Scalar Selection Mechanisms for Multi-Objective Optimization. In: *2010 IEEE Congress on Evolutionary Computation (CEC'2010)*, Barcelona, Spain, IEEE Press (2010)
3. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6** (2002) 182–197
4. Zhang, Q., Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* **11** (2007) 712–731
5. Li, H., Zhang, Q.: Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation* **13** (2009) 284–302
6. Talbi, E.G., Mostaghim, S., Okabe, T., Ishibuchi, H., Rudolph, G., Coello Coello, C.: Parallel approaches for multi-objective optimization. In Branke, J., Deb, K., Miettinen, K., Slowinski, R., eds.: *Multiobjective Optimization*. Volume 5252 of *Lecture Notes in Computer Science*. Springer (2008) 349–372
7. Nebro, A., Luna, F., Talbi, E.G., Alba, E.: Parallel Multiobjective Optimization. In Alba, E., ed.: *Parallel Metaheuristics*. Wiley-Interscience, New Jersey, USA (2005) 371–394 ISBN 13-978-0-471-67806-9.
8. Kamiura, J., Hiroyasu, T., Miki, M., Watanabe, S.: MOGADES: Multi-objective genetic algorithm with distributed environment scheme. In: *Computational Intelligence and Applications (Proceedings of the Second International Workshop on Intelligent Systems Design and Applications: ISDA'02)*. (2002) 143–148
9. Streichert, F., Ulmer, H., Zell, A.: Parallelization of Multi-objective Evolutionary Algorithms Using Clustering Algorithms. In Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E., eds.: *Evolutionary Multi-Criterion Optimization*. Third International Conference, EMO 2005, Guanajuato, México, Springer. *Lecture Notes in Computer Science* Vol. 3410 (2005) 92–107
10. Zaharie, D., Petcu, D.: Adaptive pareto differential evolution and its parallelization. In Wyrzykowski, R., Dongarra, J., Paprzycki, M., Wasniewski, J., eds.: *PPAM*. Volume 3019 of *LNCS*., Springer (2003) 261–268