**CONTI'2006**
The 7th INTERNATIONAL CONFERENCE ON TECHNICAL INFORMATICS, 8 - 9 June 2006, TIMISOARA, ROMANIA

# Communication Strategies in Distributed Evolutionary Algorithms for Multi-objective Optimization

Daniela Zaharie [*] and Dana Petcu [**]

[*] Department of Computer Science, West University of Timişoara, Faculty of Mathematics and Computer Science, bv. Vasile Pârvan, nr. 4, 300223 Timişoara, Romania
Phone: (40) 256-592157, Fax: (40) 256-592316, E-Mail: dzaharie@info.uvt.ro, WWW: http://www.info.uvt.ro/~dzaharie

[**] Department of Computer Science, West University of Timişoara, Faculty of Mathematics and Computer Science, bv. Vasile Pârvan, nr. 4, 300223 Timişoara, Romania
Phone: (40) 256-592157, Fax: (40) 256-592316, E-Mail: petcu@info.uvt.ro, WWW: http://www.info.uvt.ro/~petcu

***Abstract* – The communication between subpopulations in a distributed evolutionary algorithm is an important issue since it influences the algorithm effectiveness in solving the optimization problem and the efficiency of the parallel implementation. Choosing the adequate communication strategy depends on various factors, thus by comparing different strategies one can collect knowledge on how to design an effective approach. The aim of this paper is to compare a set of communication strategies both with respect to their effectiveness in approximating the Pareto set of a multi-objective optimization problem and with respect to the efficiency of a parallel implementation.***

***Keywords:* multi-objective optimization, evolutionary algorithms, parallel computing, island model, communication strategy.**

## I. INTRODUCTION

Many optimization problems in engineering are of multi-objective type, meaning that one have to optimize simultaneously several, usually conflicting, objectives. Dealing with multiple conflicting criteria leads to the necessity of accepting compromise solutions. In mathematical programming such compromise solutions are called Pareto optimal solutions. The notion of Pareto optimality is defined based on the concept of non-dominance. In the case of a minimization problem a vector of objective values is considered to be non-dominated if does not exist another vector having all components smaller than the corresponding components of the first vector and at least one component strictly smaller. The aim of multi-objective optimization methods is to find a set of reciprocally non-dominated vectors, called Pareto front. The corresponding set of decision vectors is called Pareto optimal set.

Evolutionary algorithms (EA) proved to be appropriate tools in approximating the Pareto optimal set. In the last decade dozens of evolutionary algorithms for multi-objective optimization (usually called multi-objective evolutionary algorithms - MOEAs) have been proposed ([3], an extensive list of references can be found at http://www.lania.mx/~ccoello/). One of the main problems of evolutionary approaches for multi-objective optimization is the computational cost. The main cause of the high computational cost is the fact that for difficult problems there is necessary to work with large populations which lead to a large number of objective functions evaluations. Besides the efforts in developing evolutionary algorithms with a better convergence rate, another way to reduce the computational cost is to use the benefits of parallel and/or distributed implementations. Parallel implementations of evolutionary algorithms have been first developed for single objective optimization [2]. Later they have been applied also for multi-objective optimization. However there are fewer studies concerning parallel and distributed MOEAs than in the case of single objective case.

There are three main parallelization models for evolutionary algorithms: master-slave, island based and cellular [2]. The most flexible is the island-based model (also called distributed model). In this parallelization model, the population is divided in several communicating subpopulations, an independent EA being executed in each subpopulation. The main advantage of this approach is its flexibility and adequacy for heterogeneous networks. The main disadvantage is that by dividing the population in subpopulations the explorative power is reduced and the approximation of the Pareto set could be poorer than in the case of a single large population. While in the case of single objective optimization by using the island model one can improve the effectiveness of the algorithm even in the case of sequential implementations, this is not always true for multi-objective problems.

There are at least two questions which should be answered when a distributed multi-objective evolutionary algorithm is designed: (i) how to assign to each subpopulation a part of the search space? (ii) what kind of information should be transferred between subpopulations and how should be realized such a transfer?

In the last years different distributed MOEA have been proposed and studied [1,5,6,7,10,11]. These studies concern mainly with different strategies of dividing the entire population in subpopulations and less on the communication strategies between subpopulations. Hence there does not exist a systematic comparative study of the influence of different communication strategies on the effectiveness and efficiency of distributed MOEAs. The aim of this paper is to present the results of such a comparison.

The paper is organized as follows. In section II is presented a summary of some distributed MOEAs with an emphasis on the particularities of their communication strategies. An analysis of four communication strategies is presented in section III. The influence of the communication on the effectiveness and on the efficiency of distributed MOEAs is illustrated in section IV through some numerical experiments both in the case of sequential and in the case of parallel implementation.

## II. DISTRIBUTED EVOLUTIONARY ALGORITHMS FOR MULTI-OBJECTIVE OPTIMIZATION

### *A. Distributed Approaches of MOEAs*

A distributed MOEA is based on the idea of dividing the population in some subpopulations. On each subpopulation an independent evolutionary algorithm is applied for a given number of generations, then a communication process is started. The distributed approach can be homogeneous (characterized by the fact that the same algorithm is applied on each subpopulation) or heterogeneous (the algorithms applied to subpopulations differ at least with respect to their parameters). Various distributed MOEAs differ with respect to one of the following elements: (i) the MOEA applied at subpopulations level; (ii) the subpopulations structure and the criteria of dividing the population in subpopulations; (iii) the communication process between subpopulations.

Most distributed MOEAs proposed in the last years are based on the NSGA II algorithm [4]. An explanation of this choice is the fact that NSGA II does not use an archive making easier the distribution of the evolutionary algorithm.

The process of dividing a population in subpopulations can be guided or not. In a non-guided approach the population is rather arbitrary divided in subpopulations and each subpopulation tries to approximate the entire Pareto front. In guided approaches each subpopulation is focused on a part of the search space. The criterion used in the division process is based either on the decision variable [10] or on the objective values [1, 5, 6]. The subpopulations are reorganized either explicitly (by periodically gathering all subpopulations and by dividing the entire set in new subpopulations [6, 10]) or implicitly by periodical exchange of information between subpopulations [1,5,7,11].

### *B. Communication strategies*

The exchange of information (elements of the population and/or control information) between subpopulations can be realized in one of the following ways: (i) by migration; (ii) by pollination; (iii) by using a shared archive.

The communication by migration consists in moving one element from a subpopulation to another subpopulation. If the subpopulations are of fixed size then in the place of the migrating element a new element is introduced (either selected from another subpopulation or even randomly generated). In the following, by random migration we shall refer the process of exchanging two randomly selected elements. The communication by pollination consists in copying one element from one population to another one. Each communication strategy is characterized by a communication topology, a selection scheme, a replacement scheme and some parameters which control the ratio of elements which are transferred from one subpopulation to another one.

The *communication topology* defines the relationship between subpopulations. There are three main topologies used in distributed MOEA: the fully connected topology (a subpopulation $S_i$ can communicate with any other subpopulation $S_j$), the ring topology (a subpopulation $S_i$ can communicate only with its neighbors $S_{i-1}$ and $S_{i+1}$, the hierarchical topology (the subpopulations are organized in a hierarchy and only subpopulations which are neighbors in this hierarchy can communicate) [7]. In the case of a fully connected topology the destination subpopulation can be chosen randomly [11] or in a systematic manner [1].

The *selection scheme* refers to the mechanism of choosing the element which will be transferred to another subpopulation. The selection criterion can be based on the properties of the subpopulation elements. The most used schemes are: the random scheme (the outgoing element is randomly selected) and the elitist scheme (is selected one of the best elements). In the case of multi-objective optimization the elitist selection means selecting one of the non-dominated elements.

The *replacement scheme* refers to the mechanism of choosing the element to be replaced in the target subpopulation. As in the case of selection the random and the elitist approaches are the most used. In the random version the element to be replaced is randomly selected while in the elitist case one of the worst elements is chosen. In the multi-objective case worst means dominated (in the case of MOEAs based on non-dominating sorting – as in NSGAII – elements having the lowest non-domination rank are chosen).

*TABLE 1. Classification of distributed MOEAs with respect to the communication process*

| *Distributed MOEA* | *Communication type* | *Topology* | *Selection* | *Replacement* | *Frequency* | *Ratio* |
|---|---|---|---|---|---|---|
| DRMOGA [6] | all subpopulations are gathered and reorganized by sorting them with respect to the objective functions | | | | 5 | 1 |
| Guided distribution [5] | pollination | fully connected (systematic) | elitist (non-dominated elements) | random | 5 | 0.3 |
| APDE [11] | migration | fully connected (random) | random | random | 25 | 0.5 |
| Cone separation [1] | pollination | dynamically constrained | constraints violation | new elements are added | 1 | variable |
| MRMOGA [7] | pollination | hierarchical | elitist from an archive | worst elements | not specified | not specified |
| Clustering DMOEA [10] | all subpopulations are gathered and reorganized by applying clustering in the decision space | | | | 1 | 1 |

There are two main communication parameters which control the effectiveness of the process: the *frequency* and the *communication ratio*. The frequency specifies the number of generations between two consecutive communication steps. The communication ratio refers to the ratio of elements which are involved in the communication process. Large communication frequencies and ratios lead to a high communication cost.

A summary of the communication particularities of some distributed MOEAs is presented in Table 1.

## III. ANALYSIS OF COMMUNICATION STRATEGIES

In this section we shall analyze the properties of four general communication strategies obtained by combining two topologies (random and ring topologies) with two communication mechanisms (random migration and elitist pollination). The communication mechanisms have an influence both on the evolution process and on the communication costs.

It is easy to see that migration does not change the global population but only the distribution of elements in subpopulations while the pollination usually decreases the global population variance but increases the averaged global fitness. This means that migration preserves the global population diversity and possible increases the diversity of some subpopulations. On the other hand the pollination could accelerate the convergence. The communication topology does not influence the population diversity. On the other hand it can influence the communication costs in the case of parallel implementations. In order to analyze the influence of the communication strategy on the communication cost we shall estimate the average number of messages to be transferred between subpopulations.

Let us consider a set of $s$ subpopulations $S_1, S_2, \ldots, S_s$ of sizes $m_1, m_2, \ldots, m_s$. Each element of a subpopulation can be involved in a communication process with a probability $p_m$ called migration probability (even in the case of a pollination process). Let $N_i$ be the average number of messages a population $S_i$ will send during a communication process.

### A. Random migration and random topology.

In this case there are two types of messages which a subpopulation send: messages containing the migrants and messages containing the elements which are sent in the place of immigrants. Since each element of a subpopulation $S_i$ can be selected as migrant with probability $p_m$ the average number of first type messages is $p_m m_i$. On the other hand the subpopulation $S_i$ can be selected with probability $1/(s-1)$ to be the destination population of migrants from any other subpopulation, $S_j$. Thus the average number of messages sent by each subpopulation $S_i$ is

$$N_i = p_m \left( m_i + \sum_{j=1, j \neq i}^{s} \frac{m_j}{(s-1)} \right) \qquad (1)$$

If all subpopulations have the same size, $m$, then $N_i = 2mp_m$. Since during the migration, $n$ decision variables and $r$ objective values are sent, each message consists of $(n+r)$ real values. The average number of messages sent by a subpopulation $S_i$ to another subpopulation $S_j$ is

$$N_{i \to j} = p_m m_i /(s-1) + p_m m_j /(s-1) \qquad (2)$$

It should be noted that all messages prepared at a given migration stage to be sent to a given subpopulation could be grouped in one longer message. However the first term in eq. (2) refers to the migrants sent to population $S_j$ and the second term refers to the elements sent to replace the immigrants from $S_j$. Since these two types of messages should be sent in two different stages of the migration process they cannot be grouped in only one message. Thus in this case each subpopulation sent at least two distinct messages.

## B. *Random migration and ring topology*

In the case of ring topology a subpopulation $S_i$ communicates only with two neighboring subpopulations: it sends migrants to $S_{i+1}$ and replacing elements to $S_{i-1}$ (all operations with indices are modulo $s$). The average number of messages sent by $S_i$ is $N_i=(m_i+m_{i-1})p_m$. When all subpopulations have the same size one obtains the same value as in the case of random topology. The difference appears in the average number of messages sent by $S_i$ to $S_j$ which is

$$N_{i\to j} = \begin{cases} 0 & \text{if } |i-j| \neq 1 \\ m_i p_m & \text{if } j = i+1 \\ m_{i-1} p_m & \text{if } j = i-1 \end{cases} \tag{3}$$

In the parallel implementation all messages which should be transferred from one population to another one can be grouped in only one package thus the process corresponding to a subpopulation $S_i$ will send two packages: one of length $m_i p_m(n+r)$ to the process corresponding to $S_{i+1}$ and one of length $m_{i-1}p_m(n+r)$ to the process corresponding to $S_{i-1}$.

## C. *Elitist pollination and random topology*

In the case of pollination, each subpopulation will send only one type of messages: those containing non-dominated elements. The number of these messages depends on the number of non-dominated elements in the subpopulation. If $k_i$ is the number of non-dominated elements in $S_i$ then the average number of messages sent by $S_i$ is $N_i=k_i p_m$ and it satisfies

$$p_m \le N_i \le p_m m_i \tag{4}$$

being clearly less than the number of messages sent in the migration case. The average number of messages transferred from a population $S_i$ to another population $S_j$ is

$$N_{i\to j} = p_m k_i / (s-1) \tag{5}$$

## D. *Elitist pollination and ring topology*

The only difference between this variant and the previous one is the fact that only neighboring subpopulations communicate and the average number of messages sent by $S_i$ to $S_j$ is

$$N_{i\to j} = \begin{cases} 0 & \text{if } |i-j| \neq 1 \\ k_i p_m \le m_i p_m & \text{if } j = i+1 \\ k_{i-1} p_m \le m_{i-1} p_m & \text{if } j = i-1 \end{cases} \tag{6}$$

The average number of packed messages (groups of messages prepared at one migration stage) transferred between the processes corresponding to $s$ subpopulations is summarized in Table 2 for all four cases analyzed above. A difference between migration and pollination which should be emphasized is the fact that in migration the subpopulations have to wait for replacements for their migrants while in the pollination case, once the subpopulation sent the migrants and received the incoming elements it can continue its evolution.

*TABLE 2. Total number of packed messages transferred between the processes corresponding to s subpopulations*

| Migration & random (A) | Migration & ring (B) | Pollination & random (C) | Pollination & ring (D) |
|---|---|---|---|
| $2s \le N \le 2s(s-1)$ | $N = 2s$ | $s \le N \le s(s-1)$ | $N = s$ |

## IV. NUMERICAL RESULTS

In order to analyze the influence of the communication strategy on the effectiveness of a distributed MOEA and on the efficiency of the parallel implementation we conducted numerical experiments on some test functions.

### A. *Experimental setup*

The MOEA algorithm which we used is an extension for multi-objective optimization of the differential evolution algorithm [8]. Its basic idea is to use the crossover operator specific to differential evolution [9] and the selection principle of NSGA II [4]: the elements of the parent and offspring populations are organized in non-domination layers and selected based on their position in these layers. The distributed variant consists on dividing the population in subpopulations of identical size and in applying the algorithm on each subpopulation. After a given number of generations a communication process is activated. All four communication strategies analyzed in the previous section have been implemented. For tests we chose two problems from the test suite introduced in [12]: ZDT4 (a deceptive problem with multiple local Pareto fronts) and ZDT6 (a problem with non-uniformly distributed Pareto front). In both cases there are two objective functions which should be minimized and n=10 decision variables. The true Pareto fronts are: $\{(x,1-\sqrt{x}), x\in[0,1]\}$ for ZDT4 and $\{(x,1-x^2), x\in[0,1]\}$ for ZDT6, respectively.

### B. *Comparative results on effectiveness*

In order to compare the quality of results produced by the four variants of communication between subpopulations we used two performance measures: a unary one (generational distance – GD) and a binary one (two set coverage – CS) [3]. The generational distance measures how far is the approximated Pareto front from the true Pareto front and it is defined as

$$GD = \frac{\left(\sum_{i=1}^{m} d^2(F_i, TF)\right)^{1/2}}{m} \tag{7}$$

where $(F_1,\ldots F_m)$ is the approximation of the Pareto front, $TF$ is the true Pareto front and $d$ is a distance from a point to a set. The two set coverage measure is used to compare two approximations of the Pareto front. If $F'$ and $F''$ are two approximations of the Pareto front then the two set coverage measure of *F'* and *F"* is defined as

$$CS(F',F") = \frac{card\{y" \in F" \mid exists\ y' \in F', s.t. y' \succ y"\}}{card(F")}$$

Thus *CS(F',F")* is the ratio of elements in *F"* which are dominated by elements in *F'*. Values of *CS(F',F")* near 1 suggest that *F'* is a significantly better approximation than *F"*.

The parameters used in implementation were: s=4 subpopulations each one having m=50 elements, a communication frequency of 50 generations and a migration probability of 0.1. The parameters corresponding to the MOEA applied to each subpopulation were chosen as in [8]: p=F=0.3. In Table 3 and 4 are presented comparative results for all communication strategies: random migration and random topology (A), random migration and ring topology (B), elitist pollination and random topology (C) and elitist pollination and ring topology (D). The values on the diagonals of the tables correspond to the generational distance (GD) while the other values correspond to two set coverage measures. All values are averages obtained through 10 independent runs of the algorithms (in fact for the two set coverage measure the averages are over 10*10 values obtained by combining 10 results for one strategy with other 10 results the other strategy).

TABLE 3. Results for ZDT6

| GD | (A) | (B) | (C) | (D) |
|---|---|---|---|---|
| (A) | *0.00309* | 0.17811 | 0.94596 | 0.89433 |
| (B) | 0.52265 | *0.00272* | 0.95598 | 0.86567 |
| (C) | 0.00373 | 0.00033 | *0.02293* | 0.87702 |
| (D) | 0 | 0 | 0.01191 | *0.05899* |

TABLE 4. Results for ZDT4

| GD | (A) | (B) | (C) | (D) |
|---|---|---|---|---|
| (A) | *0.00013* | 0.05145 | 0.44864 | 0.65358 |
| (B) | 0.00390 | *0.00005* | 0.25025 | 0.52249 |
| (C) | 0.00280 | 0.00035 | *0.01063* | 0.69227 |
| (D) | 0.00120 | 0.00015 | 0.14222 | *0.10372* |

The GD values suggest that for both test problems the variant using random migration leads to a better approximation than the variant based on elitist pollination. The values of two set coverage measure also show that the random migration lead to better results than the elitist pollination. In the case of ZDT6 test problem more than 90% of the Pareto front approximated by the variant with elitist pollination is dominated by the Pareto front approximated with the variant based on random migration. The difference between the approximations of Pareto front is also illustrated in fig. 1. In the case of ZDT4 the superiority of random migration is not so obvious. However the number of situations when a poor approximation is obtained is larger in the case of elitist pollination than in the case of random migration (see fig. 2). Concerning the difference between random topology and ring topology the experiments did not prove a clear superiority of one strategy over the other. However when is combined with random migration the ring topology seems to behave better than the random topology while when it is combined with elitist pollination it behaves worse. A possible explanation of this behavior is that the random migration ensures the diversity of population thus is no need for a random topology while in the case of elitist pollination the random topology brings the variability which is not ensured by pollination.

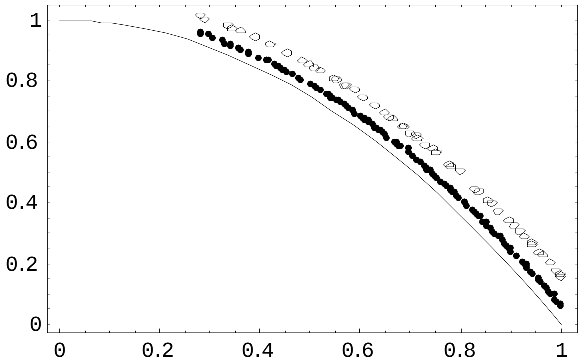


Fig. 1. Approximations of the true Pareto front for the test problem ZDT6 (continuous line): random migration & random topology (filled points) vs. elitist pollination & random topology (empty points)

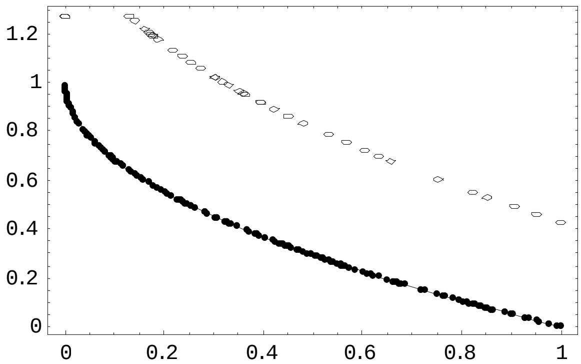


Fig. 2. Approximations of the true Pareto front for the test problem ZDT4 (continuous line): random migration & ring topology (filled points) vs. elitist pollination & ring topology(empty points)

### *C. Comparative results on communication costs*

In order to analyze the influence of the communication strategy on the efficiency of a parallel implementation (based on PVM communication library) we conducted some tests on a cluster consisting of 8 nodes (Intel P4, 3GHz, 1 Gb RAM) connected at 100Mbps. Let *T(p,s)* be the running time corresponding to the case when the population is divided in *s* subpopulations and these subpopulations are assigned to *p* processors. Each processor will execute the evolutionary algorithm corresponding to *s/p* subpopulations. The influence of the number of processors and of the communication strategy on the speedup ratio *T(1,s)/T(p,s)* is illustrated in fig.3 (for s=24 subpopulations, each one of 10 elements, a communication frequency of 5 generations, 100 communi-

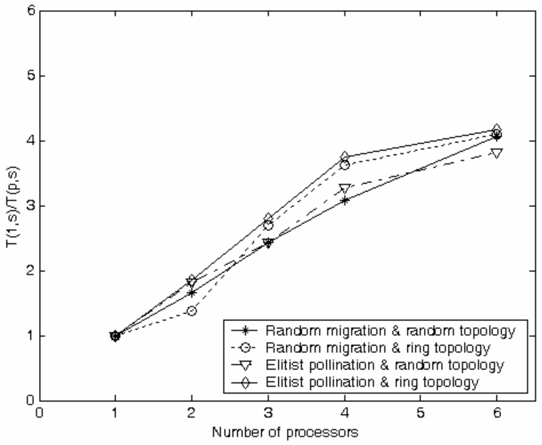


Fig. 3. Dependence of *T(1,s)/T(p,s)* on the number of processors *p* for different communication strategies ( $s \geq p$ )

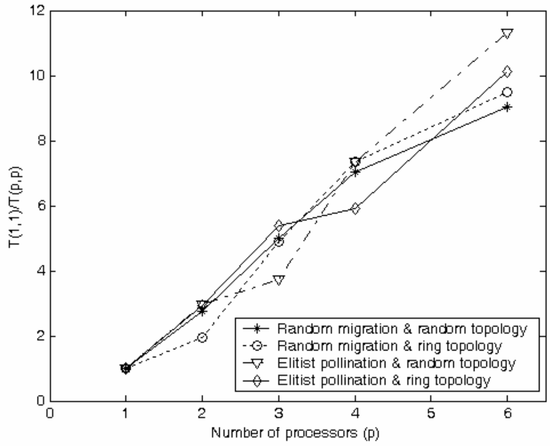


Fig. 4. Dependence of *T(1,1)/T(p,p)* on the number of processors *p* for different communication strategies.

cation steps and a migration probability equal to 0.1). No significant differences are between different communication strategies even if the number of transferred messages is higher in the case of random migration than in the case of elitist pollination. These can be explained by the fact that elitist pollination involves some extra non-domination analysis (in order to reorganize the non-domination layers after the receipt of a new element). Fig. 4 illustrates similar results but with respect to the ratio T(1,1)/T(p,p). In this case the number of subpopulations is identical with the number of processors, each processor executing only one algorithm. The superlinear behavior can be explained by the fact that the global population size is constant (240 elements). This means that when there are few subpopulations they have larger sizes than in the case of many subpopulations. Since the non-domination sorting step is of quadratic complexity with respect to the subpopulation size this leads to higher costs in the case of fewer (and implicitly larger) subpopulations.

## V. CONCLUSIONS

With respect to the ability of approximating the Pareto front, the communication based on random migration seems to be superior to that based on elitist pollination. With respect to the communication costs the elitist pollination and the ring topology are superior to their random counterparts. However the comparative study based on a parallel implementation suggests that the efficiency gain of the elitist pollination with respect to random migration is not significant. Thus, even if random migration is rarely used in current distributed MOEAs it can be considered as a viable alternative for communication between subpopulations.

## ACKNOWLEDGMENTS

This work is partially supported by the project CEEX 95/2005-2008.

## REFERENCES

[1] J. Branke. H. Schmeck, K. Deb, R.S. Maheshwar, “Parallelizing multi-objective evolutionary algorithms: cone separation“, in Proceedings of the Congress on Evolutionary Computation, IEEE Press, pp. 1952-1957, 2004.

[2] E. Cantu-Paz, “A survey of parallel genetic algorithms”, in Calculateurs Paralleles, Reseaux et Systems Repartis. Paris: Hermes,Vol. 10, No. 2. pp. 141-171,1998.

[3] C.A. Coello Coello, D.A. van Veldhuizen, G.B. Lamont, Evolutionary algorithms for solving multi-objective problems, Kluwer Academic Publishers, 2002.

[4] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, “A fast and elitist multi-objective genetic algorithm: NSGA-II”. IEEE Transaction on Evolutionary Computation, vol. 6, no. 2, pp. 181-197, 2002.

[5] K. Deb, P. Zope, A. Jain, “Distributed computing of Pareto-optimal solutions using multi-objective evolutionary algorithms”, Proc. of the second Evolutionary Multi-criterion Optimization Conference (EMO-03), 8-11 April, Faro, Portugal, LNCS 2632, pp. 535-549, 2003.

[6] T. Hiroyasu, M. Miki, S. Watanabe, “The new model of parallel genetic algorithm in multi-objectiveoptimization problems - divided range multi-objective genetic algorithm”, in *2000 IEEE Congress on Evolutionary Computation (CEC'2000)*, pp. 333--340, IEEE Press, Vol. 1,2000.

[7] A. L. Jaimes and C. A. Coello Coello, “MRMOGA: Parallel Evolutionary Multiobjective Optimization using Multiple Resolutions”, in *2005 IEEE Congress on Evolutionary Computation (CEC'2005)*, pp. 2294--2301, IEEE Press, Vol. 3, Edinburgh, Scotland, September 2005.

[8] N. K. Madavan, “Multiobjective Optimization Using a Pareto Differential Evolution Approach”, in *Congress on Evolutionary Computation (CEC'2002)*, Vol. 2, pp. 1145--1150, IEEE Service Center, Piscataway, New Jersey, May 2002.

[9] R.Storn, K. Price, “Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces”, Techn. Rep. TR-95-012, ICSI, 1995.

[10] F. Streichert, H. Ulmer, A. Zell, „Parallelization of multi-objective evolutionary algorithms using clustering algorithms“,in Carlos A. Coello Coello and Arturo Hernandez Aguirre and Eckart Zitzler (editors), *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005*, pp. 92--107, Springer. Lecture Notes in Computer Science Vol. 3410, Guanajuato, Mexico, March 2005

[11] D. Zaharie, D. Petcu, “Adaptive Pareto differential evolution and its parallelization”, in R. Wyrzykowski et al. (eds), Proc. 5th Intern. Conference PPAM2003, LNCS 3019, 261-268, 2004.

[12] E. Zitzler, K. Deb, L. Thiele, „Comparison of multiobjective evolutionary algorithms“, IEEE Transactions on Evolutionary Computation, 8(2), pp. 125-148, 2000.