論　文

MOEA/D-RAD - Resource Allocation by Diversity

Yuri Lavinas — University of Tsukuba, Graduate School of Systems and Information Engineering, Japan
yclavinas@gmail.com

Claus Aranha — (affiliation as previous author)
caranha@cs.tsukuba.ac.jp

keywords: resource allocation, diversity assessment, multiobjective optimization, priority functions.

Summary

MOEA/D decomposes multi-objective problems into single-objective subproblems and solve them in a parallel. In standard MOEA/D, all subproblems receive the same computational effort. However, as each subproblem relates to different areas of the objective space,it is expected that some subproblems are more difficult than others. Using Resource Allocation, MOEA/D could spend less effort on easier subproblems and more on harder ones, improving efficiency. in this paper, ew address Resource Allocation by using priority functions. Therefore, we propose the MOEA/D-RAD, a MOEA/D that considers diversity in the decision space as priority function. We compare MOEA/D-RAD, MOEA/D-DE and MOEA/D-DRA on the BiBBOB benchmark. This benchmark is composed of 55 BiBBOB functions grouped into 15 function classes, given similarity properties. We investigate the performance of MOEA/D-RAD, MOEA/D-DE and MOEA/D-DRA in all of these 15 groups. Exploratory experiment results shows that MOEA/D-RAD outperforms its peers on 24 functions in terms of the hypervolume metric and indicates that MOEA/D-RAD has high hypervolume values in 5 group classes. Among these 5 groups, it performs well in the moderated- and weakly-structured classes. These results validate the effectiveness of using diversity in the objective space as priority function in the MOEA/D framework.

1. Introduction

Multi-objective Optimization Problems (MOP) are maximization (or minimization) problems characterized by multiple, conflicting objective functions. It arises in real world applications that require a compromise among multiple objectives.

$$\text{minimize} f(x) = (f_1(x), ..., f_m(x)), x \in \mathbb{R}^D, \quad (1)$$

where m is the number of objective functions and $\mathbb{R}^m$ is the objective function space. $x \in \mathbb{R}^D = \{x_1, x_2, ..., x_D\}$ is a D-dimensional vector which represents a candidate solution with D variables, $f : \mathbb{R}^D \to \mathbb{R}^m$ is a vector of objective functions.

These objectives often conflict with each other, as there is no point in $\mathbb{R}^D$ that minimizes all the objectives at the same time. Consequently, the goal of the MOP optimization algorithm is to find the approximate set of solutions that balance the different objectives in an optimal way.

This balance is defined by the concept of "pareto dominance". Given two solutions vectors u, v in $\mathbb{R}^D$, u Pareto-dominates v, we say that denoted by $f(u) \prec f(v)$, if and only if $f_k(u) \leq f_k(v), \forall_k \in \{1, ..., m\}$ and $f(u) \neq f(v)$. Likewise, a solution $x \in \mathbb{R}^D$ is considered Pareto-Optimal if there exists no other solution $y \in \mathbb{R}^D$ such that $f(y) \succ f(x)$, i.e., if x is non-dominated in the feasible decision space. A non-dominated solution exists if no other solution provides a better trade-off in all objectives. Consequently, the set of all Pareto-Optimal solutions is known as the Pareto-Optimal Set (PS), while the image of this set is referred to as the Pareto-optimal Front (PF).

$$PS = \{x \in \mathbb{R}^D | \nexists y \in \mathbb{R}^D : f(y) \succ f(x)\}, \quad (2)$$

$$PF = \{f(x) | x \in PS\}. \quad (3)$$

The Multi-Objective Evolutionary Algorithm Based on Decomposition (MOEA/D) [Zhang 07] is an effective algorithm for solving MOPs. The key idea of the MOEA/D is that the multi-objective optimization problem is decomposed into a set of single objectives subproblems All subproblems are then solved in parallel.

In the original MOEA/D, all subproblems are treated uniformly, in the sense that all of them receive the same computational effort. However, it has been observed that some subproblems are harder than others, and take more effort to converge to an optimal solution [Zhou 16]. Because of this, *Resource Allocation* approaches have been proposed to allocate different amount of computational effort to different subproblems, based on an estimation of

∗1　現職:進化計算研究所

the relative difficulty of each subproblem [Zhou 16], [Zhang 09], [Kang 18]. The most popular MOEA/D algorithm related to Resource Allocation is the MOEA/D-DRA [Zhang 09]. It uses the Relative Improvement for estimating subproblem difficulty, which calculates how much a subproblem has improved in recent iterations.

Here, we propose a new approach for estimating difficulty and calculating priority in Resource Allocation for MOEA/D. Our approach uses the idea of *diversity* in objective space to calculate the priority of solutions. Our motivation for this choice is that the quality of a MOP solution set is often evaluated by the diversity in the objective space. If we assign higher priority for regions with lower diversity, we are encouraging the algorithm to spend more computational effort in regions that are not yet well explored.

In this paper, we define a priority function based on diversity on the objective space using the MRDL, proposed by Gee [Gee 15]. The MRDL is an online diversity metric based on a geometrical perspective and indicates the loss of diversity related to a solution to the whole population. We understand that this priority function is able to monitor diversity during the execution of the algorithm guiding the search behavior of the algorithm.

We compare the new approach with the MOEA/D-DRA and with the standard MOEA/D (with no priority function). The results show that a priority function focused on decision space lead to better results on the metric Hypervolume (HV).

2. Related Works

2·1 Priority functions

We define priority functions (also called utility functions) as one way of establishing preferences [Chankong 83] between solutions for resource allocation. These functions are used to decide how to allocate computational resources among subproblems by monitoring the algorithm search and guiding the distribution over iterations [Cai 15].

Only a few studies have been concerned with resource allocation. We highlight two groups. The first is composed by MOEA/D-GRA [Zhou 16], MOEA/D-DRA [Zhang 09] and in the Two-Level Stable Matching-Based Selection in MOEA/D [Nasir 11]. The other is composed by EAG-MOEA/D [Cai 15] and MOEA/D-CRA [Kang 18].

According to Zhou and Zhang [Zhou 16], MOEA/D-GRA could be seen as an extension of MOEA/D-DRA and MOEA/D-AMS [Chiang 11]. They reason that all these algorithms use a very similar priority function and that MOEA/D-GRA can simulate the behavior of MOEA/D-DRA or MOEA/D-AMS by changing the values of a single parameter.

This priority function is named as the Relative Improvement (R.I.) and defines the priority values of each subproblem $i = 1, ..., N$, as

$$\delta_i = \frac{\text{old function value} - \text{new function value}}{\text{old function value}}. \qquad (4)$$

Where new function value is the value at the current iteration (T) and old function value is the value at iteration ($T - \Delta T$).

For MOEA/D-GRA $u_i = \delta_i$, but in MOEA/D-DRA, as well as in the Two-Level Stable Matching-Based Selection in MOEA/D, a second equation is used,

$$u_i = \begin{cases} (0.95 + 0.05 \cdot \frac{\delta_i}{0.001} \cdot u_i), & \text{if } \delta_i > 0.001, \\ 1, & \text{otherwise} \end{cases}$$

The R.I. is based the assumption that if a subproblem has been improved over the last ΔT iteration (*old function value*), it should have a high probability of being improved over the next iterations.

The priority function used EAG-MOEA/D [Cai 15] and MOEA/D-CRA [Kang 18] differ from the ones in the MOEA/D-GRA group. In their case, the framework keeps two populations: one working population, and one external archive. This priority function estimates priorities for a subproblem given the number of solutions from that subproblem that are in the external archive.

Together these studies indicate that it is worth monitoring the algorithm behavior and guiding its search, but it is unclear how the choice priority functions influence the results since in all the impact of using priority functions was not isolated. In all Resource Allocation works mentioned above the choice of priority function was just one of multiple changes applied to the base framework. For example, in Zhang et al. used a 10-tournament selection in MOEA/D-DRA [Zhang 09], while Zhou and Zhang used a new replacement strategy in MOEA/D-GRA [Zhou 16]. Chiang in MOEA/D-AMS proposes an adaptive mating selection mechanism as dynamically adjusts the mating pools of individuals [Chiang 11]. Finally, both the studies of Cai and Lai in EAG-MOEA/D [Cai 15] and Kang et al. in MOEA/D-CRA [Kang 18] used an archive population.

We focus on improving the performance of the standard MOEA/D by using priority functions, given the recent success of addressing Resource Allocation with priority functions. We believe in the idea that diversity is a critical issue in the search process for any multi-objective algorithm. Therefore, we consider using priority functions to address lack of diversity aiming to make solutions better spread among each other. The proposed priority function focus on how to better spread solutions the Pareto Set (diversity on the decision space). For that, we define the MRDL priority function.

2·2 Diversity Metric

One way to measure diversity is to use metrics that evaluate MOPs solvers. The hypervolume indicator (HV) [Zitzler 98] and the Inverted Generational Distance (IGD) [Zhang 08] are frequently used as metrics to evaluate such solvers. However they include information about both quality of the solutions and diversity in a single metric.

Among the metrics that only measure diversity, there are mainly two groups. The offline group, that calculate the diversity after the execution of the algorithm, while online group, that calculate the diversity during the execution of the algorithm. We are interested in measuring diversity during the execution of the algorithm, therefore we briefly introduce some studies that are part of the online group.

The online group includes: sigma method [Mostaghim 03] (PF lies in the positive objective space); entropy of the solutions by using Parzen window density estimation-[Tan 08] (sensitive to kernel width); and maximum relative diversity loss, MRDL, [Gee 15] (expensive $O(N^2)$, with N being the size of the parent population).

In this work we chose to apply the MRDL as the strategy to measure diversity on the objective space. This is an online diversity metric estimates the diversity loss of a solution to the whole population [Gee 15]. High values indicate the existence of similar solutions or that the offspring solution is close to the convergence direction. The further an objective vector of a solution is from the convergence direction, the more it contributes for the diversity of the approximated the Pareto Front. The MRDL is the maximum value for Relative Diversity Loss (RDL) of each solution.

3. Proposed Method

In this work we propose a variant of the MOEA/D, the MOEA/D with online Resource Allocation by Diversity Metric (MOEA/D-RAD). This algorithm uses the maximum relative diversity loss, MRDL, for determining the values of the priority function.

MOEA/D-RAD is described in algorithm 1. This basic algorithm is similar to the MOEA/D-DE [Zhang 09] with exception of lines 4 and 7. Line 4 deals with the selection

Algorithm 1 MOEA/D-RAD

```
1: Initialize the weight vectors λ_i, the neighborhood B_i,
   the priority value u_i every subproblem i = 1, ..., N.
2: while Termination criteria do
3:     for 1 to N do
4:         if rand() < u_i then
5:             Generate an offspring y for subproblem i.
6:             Update the population by y.
7:     Evaluate and after ΔT iterations, keep updating u
   by a priority function.
```

of solutions given their priority function values, while the line 7 deals with the calculation of the priority function values. All other procedures and parameters are the same as in MOEA/D-DE [Li 09]. We highlight that the neighborhood is only calculated in the initialization period.

The selection of priority functions provides an important way to control MOEA/D. They allow ways of designing MOEA/D variants that might focus on desired characteristics, such as diversity, performance contribution, convergence to a specific region of the PF or others. This is possible because different methods can be used as priority functions to create the vector u in algorithm 1.

We initialize the value of the vector $u = 1$, as in MOEA/D-DRA. As in DRA and GRA we have a learning period of ΔT iterations. Here $\Delta T = 20$ as in MOEA/D-GRA [Zhou 16]. A sensitivity analysis should be performed for deciding suitable initial values for u and for ΔT.

It should also be noted that if the priority function values results in less than 3 subproblems being updated in one iteration, we reset the priority vector $u = 1$ and all subproblems will be chosen for offspring reproduction at the that iteration.

3·1 Priority Function - MRDL

To consider diversity on the objective space, we propose a priority function based on the the Maximum Relative Diversity Loss, MRDL [Gee 15].

The diversity on objective space as a priority function is based on the Maximum Relative Diversity Loss, MRDL [Gee 15]. The idea of using MRDL is that by measuring diversity on the objective space, more resources are given to incumbent solutions that have similar objective function values between two consecutive iteractions. Therefore, it is expected that this will lead to a higher exploration of the objective space. Algorithm 2 gives the details on implementation.

Algorithm 2 MRDL

```
Input: old MRDL (initial value is 0); Y^t, objective function values from the incumbent solutions; Y^{t-1}, objective function values from the incumbent solutions of the previous iteraction; N, the population size.
for i=1 to N do
    find index h where (Y_h^{t-1} ⪰ Y_i^t) and ||Y_h^{t-1} − Y_i^t|| is minimal.
    if If none is found then
        MRDL[i] = −∞
    else
        d.conv = Y_i^t − Y_h^{t-1}.
        for j=1 to N do
            p′ = Y_j^{t-1} − Y_h^{t-1}
            c′ = Y_j^t − Y_i^t
            proj_{d.conv} * p′ = sum(conv · p′) / (p′ × p′) * p′
            proj_{d.conv} * c′ = sum(conv · c′) / (c′ × c′) * c′
            RDL_j = ||p′ − proj_{d.conv} p′|| / ||c′ − proj_{d.conv} c′||
        MRDL[i] = maximum RDL
u = 1 - scale (MRDL - old MRDL) // between 0 and 1
return u, MRDL
```

The calculation of MRDL depends on the concept of weak dominance [Zitzler 03]. A solution a weakly dominates b if in all objectives $a \geq b$ (note that $a \succeq a$).

Let N be the number of incumbent solutions and the objective values of iteraction t be Y^t and the objectives values of iteraction $t-1$ be Y^{t-1}. For each incumbent solution i, find index $h \in Y^{T-1}$. This index is the index of a parent that weak dominates the solution i. If h is not found (no parent weak dominates the solution) the MRDL value for this solution is set to $-\infty$. Given i and h, for each subproblem, the value of Relative Diversity Loss (RDL) is given by

$$RDL = \frac{||p\prime - proj_{d.conv}p\prime||}{||c\prime - proj_{d.conv}c\prime||}. \tag{5}$$

RDL is a diversity measurement quantity that indicates the amount of diversity loss of an individual solution between two consecutive iterations. High values of RDL imply a reduction of the solution spread, since the further an objective vector of a solution is from the convergence direction, the more it contributes in terms of diversity in the objective space [Gee 15]. The maximum value of RDL is the MRDL of the solution i.

4. Experimental Design

To examine the effects of MOEA/D-RAD we perform a comparative experiment of BiBBOB benchmark functions. In this experiment, we use the MOEA/D-DE implemented by the MOEADr package [Campelo 18], modified to include Resource Allocation as described in the previous section. We compare three different algorithms MOEA/D-DE and MOEA/D-DRA as well as the proposed MOEA/D-RAD.

4·1 Target Problems

The Black-Box Optimization Bi-Objective Benchmark (bbob-biobj) test functions [Tusar 16] are used as our benchmark problem sets. This test suit is composed of 55 bi-objective functions combined into 15 different classes.

We list below the function classes:

- Group 1: separable - separable;
- Group 2: separable - moderate;
- Group 3: separable - ill-conditioned;
- Group 4: separable - multi-modal;
- Group 5: separable - weakly-structured;
- Group 6: moderate - moderate;
- Group 7: moderate - ill-conditioned;
- Group 8: moderate - multi-modal;
- Group 9: moderate - weakly-structured;
- Group 10: ill-conditioned - ill-conditioned;
- Group 11: ill-conditioned - multi-modal;
- Group 12: ill-conditioned - weakly-structured;
- Group 13: multi-modal - multi-modal;
- Group 14: multi-modal - weakly structured;
- Group 15: weakly-structured - weakly-structured.

Here we give a simple explanation of the classes. A separable function does not show any dependencies between the variables while a multi-modal function have at least two minima. Moderate- and ill-conditioned have a high sensitivity in the contribution of a solution to the objective function value [Hansen 11]. Finally, a weakly-structured function is a function that the general structure is very unclear [Finck 10].

4·2 Experimental Parameters

We use the conventional MOEA/D-DE parameters [Li 09] for each Resource Allocation strategy: update size $nr = 2$, neighborhood size $T = 20$, and the neighborhood search probability $\delta_p = 0.9$. The DE mutation operator value is $phi = 0.5$. The Polynomial mutation operator values are $\eta_m 20$ and $p_m = 0.03333333$. The decomposition function is Simple-Lattice Design (SLD), the scalar aggregation function is Weighted Sum (WS), the update strategy is the Restricted Update Strategy and we performed a simple linear scaling of the objectives to [0, 1].

For every strategy/function pair we perform 21 repetitions with 30000 function evaluations and population size $N = 150$.

4·3 Experimental Evaluation

We compare the results of the different strategies based on their Hypervolume (HV) metric. Higher values of the HV indicate better approximations of the Pareto Front.

For the calculation of HV, the objective function was scaled to the $0, 1$ interval, and the reference point was set to $(1, 1)$. To verify any statistical differences in the results for the different strategies, we use the Pairwise Wilcoxon Rank Sum Tests with confidence interval $\alpha = 0.05$ and with the Hommel adjustment method for multiple comparisons.

5. Results

Figure 1 show box-plots of the HV values from experiments with MOEA/D-DE, MOEA/D-DRA and MOEA/D-RAD on BiBBOB functions F26 (moderate and weakly-structured) where MOEA/D-RAD performs the best, F1 (separable and separable) where MOEA/D-DRA performs the best and F32 (moderate and multi-modal) where MOEA/D-DE performs the best.

In all Figures and in Table 1, we can see that, in general, using MOEA/D-RAD is a good choice, being the best or the second best. However this depends on the characteristics of the function. Table 1 reinforces these results. In this exploratory experiment, it came to our understanding that the algorithms may find it easier to solve similar group of functions (for more information, see Section 4·1).

MOEA/D-RAD might perform well on the functions where at least one objective is in one of the following classes: moderate-conditioned or weakly-structured. while MOEA/D-DE seems to perform better in functions that are multi-modal. A more careful experiment should be conducted to verify these hypotheses.

In summary, these results show that MOEA/D-RAD perform better than MOEA/D-DRA, showing an improvement over resource allocation algorithms., and also outperforms MOEA/D-DE, in 24 functions. MOEA/D-DE performs the best in 19 functions while MOEA/D-DRA is the best in only 12 functions. These results are indicated by Table 1 and Table 2.

5·1 Proportion of Non-dominated Solutions

Another difference that we see among the algorithms is the proportion of non dominated solutions. The results on the Table 1 indicate that MOEA/D-DRA leads to the highest rate of non-dominated solutions in the final solution set, followed by MOEA/D-RAD. In general, both substantially improve this proportion over the results of MOEA/D-DE. It seems that there is no relation in hypervolume performance and the proportion of non-dominated solutions.

5·2 Resource Allocation

Figure 2 and Figure 3 illustrate the amount resource allocated by MOEA/D-DRA and MOEA/D-RAD to every subproblem on bbob-biobj-26 and bbob-biobj-32 problems. We show images from these functions due to space limitations. We exclude the visuals from MOEA/D-DE since it give the same amount of resource to every problem (200).

It is clear from the Figures that MOEA/D-RAD and MOEA/D-DRA influence differently the Resource Allocation. It is also important to highlight that, during the execution of the algorithm, the Resource Allocation done by MOEA/D-RAD to each subproblem is similar and not very far from 200, the default amount as well as the Resource Allocation done by MOEA/D-DE. The Resource Allocation done by MOEA/D-DRA depends on the problem indicating a more adaptive behavior.

6. Conclusion

The aim of the present research was to investigate how the proposed algorithm, MOEA/D-RAD performs in the BiBBOB benchmark functions. This algorithm differs from the standard MOEA/D by using Resource Allocation techniques computational effort proportional to each subproblem's difficulty.

MOEA/D-RAD uses a priority function based on a geometrical perspective, MRDL, for Resource Allocation. This algorithm determines the computational resource assigned to each subproblem, based on its contribution to the overall diversity of the population.

We have compared the new approach with the MOEA/D-DE and a variant with dynamic resource allocation strategy, MOEA/D-DRA, and the experimental results suggested our method performs well on many test problems.

This study has shown that MOEA/D-RAD out-performs MOEA/D-DE and MOEA/D-DRA, since it achieved high HV values in many functions. Although MOEA/D-DRA lead to the highest rate of non-dominated solutions in the

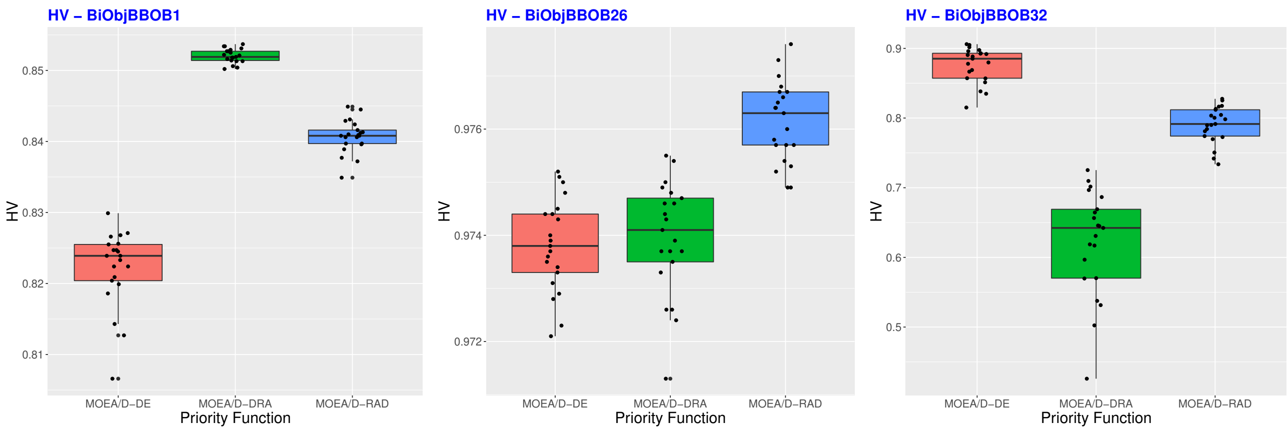

Fig. 1: Box plot of HV values on bbob-biobj-1, bbob-biobj-26 and bbob-biobj-32. (Higher values are better)

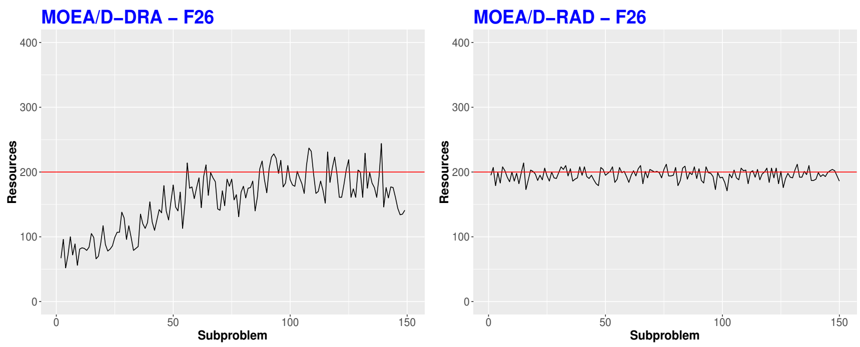

Fig. 2: Resource Allocation by subproblem - The red line indicates the default amount of resource for each problem, i.e., with no priority function.

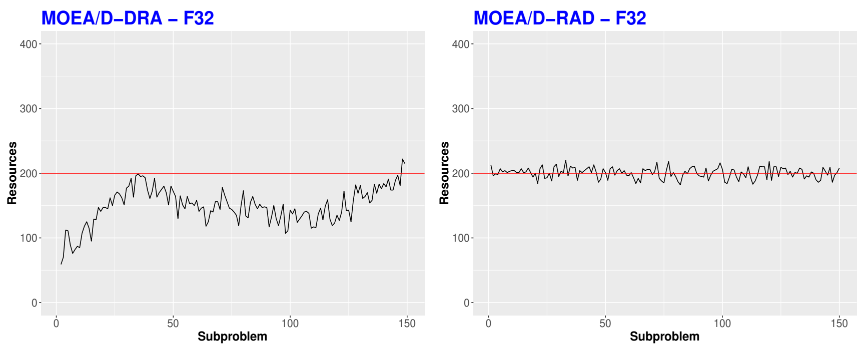

Fig. 3: Resource Allocation by subproblem - The red line indicates the default amount of resource for each problem, i.e., with no priority function.

final solution set, MOEA/D-RAD could be seem as an improvement to the MOEA/D-DE proportion values.

Since the Resource Allocation distribution of MOEA/D-RAD is not so much different to the one of MOEA/D-DE we understand that this explains why the results in the hypervolume metric of the MOEA/D-RAD tend to be better, but not so much different to the hypervolume values of the MOEA/D-DE. Consequently, we conclude that MOEA/D-RAD does represent a improvement in diversity not only since it improves the hypervolume metrics in many functions but since for *all* functions the proportion of non-dominated solutions is always higher.

Overall, the findings of this work strengthens the idea that Resource Allocation techniques are worth of attention, specially those that focuses on critical issues (such as diversity in the object space). This suggests the choice of priority functions is a critical component of a Resource Allocation system. Our results indicate that MOEA/D-RAD might be a reasonable choices MOP where at least one objective is in one of the following classes: separable; moderate-conditioned; and ill-conditioned. It is crucial to explore this further.

In this work, we do not yet consider archive based Resource Allocation and archive based priority functions, such as MOEA/D-CRA [Kang 18]. We will address this issue

Metric	HV			Proportion of Non-dominated		
Algorithm:	MOEA/D-DE	MOEA/D-RAD	MOEA/D-DRA	MOEA/D-DE	MOEA/D-RAD	MOEA/D-DRA
Group 1 - F1.	0.8239 (0.005)	0.8408 (0.002)	**0.8519 (0.001)**	0.460 (0.07)	0.910 (0.09)	1.000 (0.00)
Group 1 - F2.	0.9480 (0.003)	**0.9572 (0.001)**	0.6435 (0.043)	0.3933 (0.05)	0.800 (0.08)	0.9933 (0.04)
Group 2 - F3.	0.9113 (0.002)	**0.9211 (0.001)**	0.5783 (0.043)	0.4800 (0.06)	0.9133 (0.12)	0.9933 (0.04)
Group 2 - F4.	0.9392 (0.002)	**0.9477 (0.002)**	0.7236 (0.018)	0.4267 (0.06)	0.8267 (0.06)	1.0000 (0.01)
Group 3 - F5.	0.6722 (0.008)	0.6937 (0.006)	**0.7024 (0.003)**	0.4867 (0.07)	0.9067 (0.07)	1.000 (0.01)
Group 3 - F6.	0.8946 (0.003)	0.9092 (0.003)	**0.9146 (0.002)**	0.4533 (0.05)	0.8467 (0.06)	1.000 (0.00)
Group 4 - F7.	0.8313 (0.008)	**0.8350 (0.018)**	0.8173 (0.017)	0.3600 (0.04)	0.8533 (0.09)	0.9933 (0.03)
Group 4 -F8.	**0.8784 (0.016)**	0.8159 (0.032)	0.7500 (0.031)	0.3533 (0.06)	0.8533 (0.11)	1.000 (0.02)
Group 5 - F9.	0.9526 (0.002)	**0.9583 (0.081)**	0.8340 (0.002)	0.4533 (0.08)	0.867 (0.06)	1.000 (0.04)
Group 5 - F10.	0.7780 (0.030)	**0.7882 (0.031)**	0.7484 (0.038)	0.5267 (0.07)	0.8667 (0.07)	1.000 (0.00)
Group 1 - F11.	0.8214 (0.003)	0.8315 (0.001)	**0.8398 (0.000)**	0.6000 (0.06)	0.9200 (0.05)	1.000 (0.00)
Group 2 - F12.	0.9945 (0.001)	**0.9948 (0.001)**	0.9876 (0.005)	0.4533 (0.08)	0.8933 (0.09)	1.000 (0.00)
Group 2 - F13.	0.9890 (0.001)	**0.9907 (0.001)**	0.9855 (0.003)	0.4200 (0.07)	0.8200 (0.11)	1.000 (0.00)
Group 3 - F14.	0.8995 (0.008)	**0.9155 (0.009)**	0.6580 (0.056)	0.48733 (0.05)	0.8800 (0.08)	0.9933 (0.01)
Group 3 - F15.	0.9881 (0.002)	**0.9903 (0.002)**	0.8981 (0.033)	0.3733 (0.06)	0.7333 (0.12)	1.000 (0.02)
Group 4 -F16.	**0.9609 (0.007)**	0.9449 (0.009)	0.8562 (0.037)	0.3533 (0.07)	0.8067 (0.09)	0.9867 (0.05)
Group 4 -F17.	**0.9752 (0.017)**	0.9042 (0.022)	0.7901 (0.053)	0.3000 (0.06)	0.7667 (0.11)	0.9733 (0.13)
Group 5 - F18.	0.9455 (0.001)	**0.9455 (0.001)**	0.8988 (0.015)	0.4733 (0.09)	0.8533 (0.09)	1.0000 (0.00)
Group 5 - F19.	0.9217 (0.025)	**0.9441 (0.057)**	0.6001 (0.182)	0.4800 (0.09)	0.8333 (0.11)	1.000 (0.05)
Group 6 - F20.	0.9857 (0.001)	**0.9869 (0.001)**	0.9868 (0.001)	0.5800 (0.09)	0.9333 (0.05)	1.0000 (0.00)
Group 6 - F21.	0.9719 (0.001)	0.9756 (0.001)	**0.9766 (0.001)**	0.4933 (0.07)	0.9000 (0.09)	1.000 (0.00)
Group 7 - F22.	0.8035 (0.006)	**0.8177 (0.005)**	0.6182 (0.031)	0.5067 (0.08)	0.8800 (0.06)	1.0000 (0.02)
Group 7 - F23.	0.9677 (0.001)	**0.9714 (0.001)**	0.7871 (0.027)	0.4933 (0.05)	0.8800 (0.08)	0.9933 (0.03)
Group 8 - F24.	**0.9674 (0.009)**	0.9408 (0.013)	0.8609 (0.031)	0.3933 (0.06)	0.7800 (0.11)	0.9800 (0.12)
Group 8 - F25.	**0.9699 (0.009)**	0.9349 (0.012)	0.8647 (0.045)	0.3200 (0.05)	0.7600 (0.11)	0.9533 (0.13)
Group 9 - F26.	0.9738 (0.001)	**0.9763 (0.001)**	0.9741 (0.001)	0.5600 (0.09)	0.9467 (0.05)	1.0000 (0.00)
Group 9 - F27.	0.9437 (0.042)	**0.9480 (0.044)**	0.6161 (0.172)	0.5467 (0.08)	0.8933 (0.06)	0.9867 (0.07)
Group 6 - F28.	0.9808 (0.001)	0.9836 (0.001)	**0.9866 (0.001)**	0.4067 (0.06)	0.8333 (0.09)	1.000 (0.00)
Group 7 - F29.	0.7314 (0.016)	**0.8589 (0.008)**	0.8403 (0.009)	0.4600 (0.08)	0.8467 (0.07)	1.000 (0.00)
Group 7 - F30.	0.9687 (0.002)	**0.9738 (0.002)**	0.8009 (0.025	0.4600 (0.06)	0.8000 (0.10)	0.9933 (0.03)
Group 8 - F31.	**0.9255 (0.036)**	0.8529 (0.036)	0.7477 (0.045)	0.3400 (0.07)	0.8000 (0.08)	0.9733 (0.13)
Group 8 - F32.	**0.8852 (0.025)**	0.7915 (0.026)	0.6423 (0.076)	0.3400 (0.06)	0.7933 (0.07)	0.9733 (0.12)
Group 9 - F33.	0.9849 (0.001)	0.9867 (0.000)	**0.9885 (0.031)**	0.4400 (0.05)	0.7867 (0.08)	1.0000 (0.00)
Group 9 - F34.	0.8744 (0.038)	**0.8776 (0.031)**	0.6388 (0.182)	0.5000 (0.11)	0.8400 (0.10)	1.0000 (0.01)
Group 10 - F35.	0.5550 (0.003	0.5831 (0.004)	**0.5956 (0.003)**	0.5067 (0.07)	0.9000 (0.07)	01.000 (0.00)
Group 10 - F36.	0.7872 (0.006)	**0.7983 (0.006)**	0.7872 (0.006)	0.4733 (0.08)	0.8733 (0.09)	0.9933 (0.01)
Group 11 - F37.	0.7367 (0.009)	0.7425 (0.010)	**0.7479 (0.013)**	0.4000 (0.05)	0.8667 (0.08)	1.000 (0.01)
Group 11 - F38.	**0.8619 (0.010)**	0.8550 (0.011)	0.8461 (0.019)	0.3133 (0.05)	0.8667 (0.07)	0.9867 (0.04)
Group 12 - F39.	0.8128 (0.005)	**0.8285 (0.004)**	0.7280 (0.015)	0.5067 (0.08)	0.9000 (0.07)	1.0000 (0.00)
Group 12 - F40.	0.4736 (0.047)	**0.5116 (0.048)**	0.3954 (0.042)	0.6333 (0.08)	0.9467 (0.03)	1.0000 (0.01)
Group 10 - F41.	0.9281 (0.0.003)	0.9368 (0.004)	**0.9480 (0.003)**	0.4133 (0.06)	0.8267 (0.10)	1.0000 (0.00
Group 11 - F42.	**0.9379 (0.010)**	0.9034 (0.021)	0.8975 (0.021)	0.3667 (0.06)	0.8467 (0.08)	0.9933 (0.03)
Group 11 - F43.	**0.8744 (0.026)**	0.7937 (0.031)	0.7410 (0.060)	0.3733 (0.07)	0.7800 (0.10)	0.9533 (0.15
Group 12 - F44.	0.9687 (0.003)	**0.9761 (0.002**	0.9032 (0.011)	0.4067 (0.06)	0.8200 (0.08)	0.9867 (0.04)
Group 12 - F45.	0.8091 (0.043)	0.8223 (0.052)	**0.8557 (0.063)**	0.5200 (0.07)	0.8867 (0.05)	1.0000 (0.01
Group 13 - F46.	**0.8817 (0.012)**	0.8428 (0.023	0.8378 (0.027)	0.2933 (0.06)	0.8467 (0.09)	1.000 (0.03)
Group 13 - F47.	**0.8208 (0.031)**	0.6803 (0.043)	0.5553 (0.071)	0.4000 (0.08)	0.8667 (0.08)	0.9933 (0.04
Group 14 - F48.	**0.9520 (0.0018)**	0.8673 (0.033)	0.7799 (0.044)	0.3400 (0.07)	0.7067 (0.13)	0.9733 (0.17)
Group 14 - F49.	**0.8967 (0.051)**	0.8730 (0.059)	0.8579 (0.068)	0.4533 (0.07)	0.8533 (0.07)	1.000 (0.03
Group 13 - F50.	**0.8794 (0.036)**	0.7626 (0.038)	0.7402 (0.031)	0.3667 (0.07)	0.8600 (0.07)	1.000 (0.01)
Group 14 - F51.	**0.9801 (0.008)**	0.9535 (0.009)	0.9218 (0.017)	0.3467 (0.06)	0.8000 (0.11)	0.9667 (0.20
Group 14 - F52.	**0.8839 (0.060)**	0.8652 (0.069)	0.8414 (0.074)	0.4067 (0.07)	0.8533 (0.10)	1.0000 (0.109)
Group 15 - F53.	0.9927 (0.000)	0.9929 (0.000)	**0.9930 (0.00)**	0.5267 (0.05)	0.8600 (0.06)	1.0000 (0.00
Group 15 - F54.	**0.8950 (0.045)**	0.8717 (0.057)	0.7629 (0.135)	0.5267 (0.08)	0.9000 (0.07)	1.000 (0.02)
Group 15 - F55.	**0.3855 (0.070)**	0.3769 (0.084)	0.3709 (0.111)	0.8867 (0.07)	0.9733 (0.02)	1.000 (0.00

Table 1: On the left side, HV medians and standard deviations, in parenthesis for every function/priority function. The best values found by a priority function are in bold. On the right side, mean of the percentage of the median values and mean of the median values of the standard deviation (in parenthesis) of non-dominated solutions for every function/priority function.

in a continuation to this study. There are many components and variants of MOEA/D and is interesting to combine the Norm priority function with the them. Then, we can further explore the relationship of priority functions

HV	MOEA/D-DRA	MOEA/D-RAD
MOEA/D-RAD	3.1e-08	-
MOEA/D-DE	3.1e-08	3.1e-08

Table 2: Statistical Analysis of the HV results based on the Pairwise Wilcoxon Rank Sum Test. It shows that exists significant difference between MOEA/D-RAD, MOEA/D-DRA and MOEA/D-DE.

based on diversity with others components and variants of the MOEA/D framework. How to define more efficient and effective utility functions for different problems is also worth further investigation (such as priority function that also consider constraints) as well as to verify the results of using priority function in other real-world problems.

♢ References ♢

[Cai 15] Cai, X., Li, Y., Fan, Z., and Zhang, Q.: An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization, *IEEE Transactions on Evolutionary Computation*, Vol. 19, No. 4, pp. 508–523 (2015)

[Campelo 18] Campelo, F. and Aranha, C.: MOEADr: Component-Wise MOEA/D Implementation, URL https://cran.R-project.org/package=MOEADr (2018), R package version 1.2.0

[Chankong 83] Chankong, V. and Haimes, Y.: Multiobjective Decision Making: Theory and Methodology North Holland, *New York* (1983)

[Chiang 11] Chiang, T.-C. and Lai, Y.-P.: MOEA/D-AMS: Improving MOEA/D by an adaptive mating selection mechanism, in *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pp. 1473–1480IEEE (2011)

[Finck 10] Finck, S., Hansen, N., Ros, R., and Auger, A.: Real-parameter black-box optimization benchmarking 2009: Presentation of the noiseless functions, Technical report, Citeseer (2010)

[Gee 15] Gee, S. B., Tan, K. C., Shim, V. A., and Pal, N. R.: Online diversity assessment in evolutionary multiobjective optimization: A geometrical perspective, *IEEE Transactions on Evolutionary Computation*, Vol. 19, No. 4, pp. 542–559 (2015)

[Hansen 11] Hansen, N., Ros, R., Mauny, N., Schoenauer, M., and Auger, A.: Impacts of invariance in search: When CMA-ES and PSO face ill-conditioned and non-separable problems, *Applied Soft Computing*, Vol. 11, No. 8, pp. 5755–5769 (2011)

[Kang 18] Kang, Q., Song, X., Zhou, M., and Li, L.: A Collaborative Resource Allocation Strategy for Decomposition-Based Multiobjective Evolutionary Algorithms, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2018)

[Li 09] Li, H. and Zhang, Q.: Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Transactions on evolutionary computation*, Vol. 13, No. 2, pp. 284–302 (2009)

[Mostaghim 03] Mostaghim, S. and Teich, J.: Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO), in *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE*, pp. 26–33IEEE (2003)

[Nasir 11] Nasir, M., Mondal, A. K., Sengupta, S., Das, S., and Abraham, A.: An improved multiobjective evolutionary algorithm based on decomposition with fuzzy dominance, in *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pp. 765–772IEEE (2011)

[Tan 08] Tan, K. C., Goh, C. K., Mamun, A., and Ei, E.: An evolutionary artificial immune system for multi-objective optimization, *European Journal of Operational Research*, Vol. 187, No. 2, pp. 371–392 (2008)

[Tusar 16] Tusar, T., Brockhoff, D., Hansen, N., and Auger, A.: COCO: the bi-objective black box optimization benchmarking (bbob-biobj) test suite, *arXiv preprint arXiv:1604.00359* (2016)

[Zhang 07] Zhang, Q. and Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE Transactions on evolutionary computation*, Vol. 11, No. 6, pp. 712–731 (2007)

[Zhang 08] Zhang, Q., Zhou, A., and Jin, Y.: RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm, *IEEE Transactions on Evolutionary Computation*, Vol. 12, No. 1, pp. 41–63 (2008)

[Zhang 09] Zhang, Q., Liu, W., and Li, H.: The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances, in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pp. 203–208IEEE (2009)

[Zhou 16] Zhou, A. and Zhang, Q.: Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms, *IEEE Transactions on Evolutionary Computation*, Vol. 20, No. 1, pp. 52–64 (2016)

[Zitzler 98] Zitzler, E. and Thiele, L.: Multiobjective optimization using evolutionary algorithms—a comparative case study, in *International conference on parallel problem solving from nature*, pp. 292–301Springer (1998)

[Zitzler 03] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Da Fonseca, V. G.: Performance assessment of multiobjective optimizers: An analysis and review, *IEEE Transactions on evolutionary computation*, Vol. 7, No. 2, pp. 117–132 (2003)

〔担当委員：XX　XX〕

Received August 1, 2010.

Author's Profile

Lavinas, Yuri (Member)

Received his Bachelor degree from the University of Brasilia - Brazil, 2016, in Computer Science. Now he is pursuing his master degree in Computer Science on the University of Tsukuba - Japan. His research interests include Evolutionary Computation and optimization as well as Neuroevolution, and their applications.

Aranha, Claus (Member)

Obtained his PhD from the University of Tokyo, Graduate School of Frontier Sciences in 2010. From 2011 to 2012, was a Post-Doctoral Researcher at the COPPE School of Engineering, Federal University of Rio de Janeiro, Brasil. From 2012, is an Assistant Professor at the Faculty of Systems and Information Engineering, University of Tsukuba. His research interests include Auto-Adaptive Systems, applications of Evolutionary Computation, Computational Creativity, Multi Agent Systems and Artificial Life.