Resource Allocation by Diversity

Yuri Lavinas

***Abstract*—Multi-objective Evolutionary Algorithm based on Decomposition, MOEA/D, decomposes a multi-objective problem into single-objective subproblems. All these subproblems are uniformly treated, i.e, there is no priority among them and are related to an area of the theoretical Pareto Front. It is expected that different areas would be more difficult to find approximations that other areas leading to an unbalanced exploration of the search space. To balance this exploration, utility functions have been successfully used over the last years. In this work an experimental investigation was conducted to analyze a new proposed utility function based on an on-line diversity metric. A preliminary experimental analysis on a group of benchmark functions and in on the problem of selecting the optimal landing site for the lunar landing exploration mission was conducted in which the proposed algorithm was compared with two MOEA/D variants, MOEA/D-DE an MOEA/D-GRA. The results validate that the proposed algorithm is very competitive.**

I. Introduction

Multi-objective Optimization Problems (MOPs) [1] are problems composed of two or more functions which must be simultaneously optimized by the same set of parameters. This composition is characterized by a set of conflicting objective functions resulting in a set of optimal compromise solutions.

These m multiple objective functions must be optimized simultaneously:

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

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 and it represents a candidate solution with D variables, $f : \mathbb{R}^D \rightarrow \mathbb{R}^m$ 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 sets of solutions that balance the different objectives in an optimal way.

Given two solutions vectors u, v in $\mathbb{R}^D$, u Pareto-dominates v, 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 balance 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).

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

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

Multi-objective evolutionary algorithms (MOEAs) are one of the most widely used groups of algorithms for finding approximations to the PF of a MOP. They are characterized by their ability to find good approximations to PF in a single run [2]. In recent years, there has been an increasing interest in studying MOEAs and with a primary concern of improving their general performance. Among MOEAs, there are three major paradigms: Pareto domination-based approaches [3], [4]; the indicator-based approaches [5], [6] and the decomposition-based approaches [7], [8].

In this study, we are interested in analyzing the Multi-objective Evolutionary Algorithm based on Decomposition framework, MOEA/D [8]. It represents a class of population-based meta-heuristics for solving Multi Objective Problems [9]. In this framework, the original multi-objective problem is decomposed into simpler, single-objective subproblems by means of scalarizations which defines the weights vectors. Each subproblem is then evaluated and its utility value is calculated by an aggregation function given the related weight vector. In the original MOEA/D, all the subproblems have the same amount of computational resource (number of iteractions).

Although researchers have not studied this problem in much detail, there have been some works that have discussed this matter. One way to address this problem is to adjust the behavior of the algorithm in an online manner to suit the problem in question [10], [11], [12], [13], [14], [15] [16]. All algorithmic components can be tunned adaptively and often feedback information is needed for these adaptation strategies.

Another way, is allocating different number of evaluations to the subproblems based on some priority function. In a few recent works, a utility function is used to prioritize resources given to subproblems that contribute more to the algorithm's search. In the works of Zhang et al. [13] and Zhou et al. [17] a utility function was proposed aiming to prioritize solutions based on a historical convergence information during different generations. Another approach was implemented in [18], where the utility function was based on the presence of a solution from the main population on a secondary population.

The aim of this study is to explore the relationship between subproblems and utility functions. Here, we propose a new methodology for defining these priority functions given a diversity metric. This studies examines the integration of an online diversity metric based on a geometrical perspective [19] as direct way to define the utility function. Unfortunately, a full discussion of diversity metrics as utility function lies beyond the scope of this study.

II. Related Works

A. Utility functions

Utility functions monitor the algorithm performance and may be used to decide how to distribute the computation resources among subproblems, guiding the search behavior of the algorithm. They are one way of defining priorities [20],[21], and may be used as one way of deciding computational resources distributions among subproblems by guiding the distribution over generations [16].

To the best of the authors knowledge, there is a relatively small body of the literature concerned with resource allocation. Few previous studies are listed here: MOEA/D-DRA [13]; MOEA/D-GRA [17]; and MOEA/D-AMS [22]. On most of them, there is moderated justification for the reason of why the utility function is defined. Also, limited explanation of why these functions help the MOEA/D search.

According to Zhou and Zhang [17], MOEA/D-GRA may be seen as an generalization of MOEA/D-DRA and MOEA/D-AMS. The reason is that all of these algorithm use a very similar utility function. In all, the utility function, $u = \{u_1, u_2, ..., u_N\}$ for every subproblem $i = 1, ..., N$, is as defined in the next equation.

$$u_i = \frac{\text{old fun val} - \text{new fun val}}{\text{old fun val}}. \tag{4}$$

The idea that supports this equation is that if a subproblem has been improved over the last ΔT generations (*old function value*), it should have a high probability to be improved over the next few generations.

In contrast, the utility function used in MOEA/D-CRA [18], is based on if of an individual from population A is selected to compose the population B. This MOEA/D variation is based on the MOEA/D-M2M [23], a variation with two populations.

Together these studies indicate that it is worth monitoring the algorithm behavior and guiding its search, but it is unclear if the utility functions proposed are the most appropriated ones. Besides it remains vague how these functions affect the algorithm results.

Consequently, in this work, we propose to define an utility function supported by a diversity metric based on a geometrical interpretation of convergence and diversity. This new function is then integrated to MOEA/D-GRA. The benefit of using MOEA/D-GRA as based is that it has a straightforward implementation, it achieved good overall results in the experimental analysis of Zhou and Zhang [17] and it is a good representative of a the class of variants of MOEA/D with resource allocation.

B. Diversity Metric

Some works in diversity metrics over the last 2 decades have successfully proposed and applied in different task on evolutionary computation.

One way to measure diversity is to use metrics that evaluate MOPs solvers that consider diversity. The hyper-volume indicator (HV) [24] and the Inverted Generational Distance (IGD) [25] are frequently used as metrics to evaluate such solvers. However they include information about both convergence and diversity in a single metric.

Among the metrics that only measure diversity, there are mainly two groups. The first one is the offline group, that calculate the diversity after the execution of the algorithm, and the second is the online group, that calculate the diversity during the execution of the algorithm. Following some studies that are part of these groups are briefly introduced. Both lists are not extensive ones.

The offline group is composed off: Chi-square-like deviation [26]; Spacing method [27]; Uniformly distribution index [28]; Entropy approach [29]; Grid diversity metric [30]; sparsity measure [31]. These published studies need knowledge of the PF or the ideal vector. While the online group is composed off sigma method [32] (PF lies in the positive objective space); entropy of the solutions by using Parzen window density estimation-[33] (sensitive to kernel width); maximum relative diversity loss [19] (expensive $O(N^2)$, with N being the size of the parent population).

In this work the maximum relative diversity loss was chosen to an online diversity assessment to measure the diversity loss caused by any individual in the population and then guide the resource allocation at each generation. Since it is a very costly metric, in future works other metrics are going to be analyzed.

III. Proposed Method

MOEA/D-RAD, MOEA/D with on-line Resource Allocation by Diversity Metric, is the variant of MOEA/D proposed in this work. This algorithm uses the maximum relative diversity loss, MRDL, for determining the values of the utility function.

Algorithm 1 MOEA/D-RAD

```
1: Initialize the weight vectors $\lambda_i$, the neighborhood $B_i$, the
   utility 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                ▷ From MOEA/D-GRA
5:             Generate a offspring $y$ for subproblem $i$.
6:             Update the population by $y$.
7:     Update u by a diversity metric after $\Delta T$ generations.
```

The algorithm 1 describe MOEA/D-RAD. Except from line 4, in which a subproblem may not be part of the group that is going to be iterated and from line 7, in which the utility function is calculated, the whole procedure is similar to the MOEA/D-DE [13]. Likewise, all reproduction procedures and parameters are the same as in MOEA/D-DE [7]. It is important to highlight that the neighborhood is only calculated in the initialization period.

The decomposition method used is the Simple-Lattice Design (SLD), the scalar aggregation function Weighted Sum (WS), the Restricted update strategy.

We initialize the value of the vector $u = 1$, as in MOEA/D-DRA. As in MOEA/D-DRA and GRA we have a learning period, ΔT generations (*old function value*). Here, the $\Delta T = 20$. A sensitivity analyzes need to be performed for deciding suitable initial values for u for ΔT.

A. Utility Function by Diversity

The utility function proposed in this work is based on the Maximum Relative Diversity Loss MRDL. Prior to scalarizing it between 0 and 1 to fit the algorithm 1 we calculate MRDL for every individual of the population. The following function describes how to calculate the utility function given the MRDL, $\Gamma^{p\to c}$, as follows.

$$u_i = \Gamma_i^{p\to c} - \Gamma_j^{p\to c}, \text{with } i = 1, ..., N. \tag{5}$$

MRDL is an online diversity metric that estimates the diversity loss of a solution to the whole population [19]. First, it is an useful metric since if a new offspring generated is identical to any offspring solution in the convergence archive, the metric value will be infinite. Second, high values of indicates the existence of similar offspring solution in the convergence archive or the offspring solution is close to the line of estimated convergence direction. Finally, it may be used as an adaptive technique to adjust parameters, as applied in its original study [19].

The idea of the MRDL is to use the space movement (convergence directions) of a solution on the objective space towards the PF. The further an objective vector of a solution is from the convergence direction, more it contributes for the diversity of the approximated PF.

To compute the MRDL, we need to estimate k convergence directions at every generation. Then we need to approximate the Relative Diversity Loss (RDL) for each of these k convergence directions. To calculate this estimative of diversity loss of a solution to the whole population, the following equation is used, considering every incumbent solution related to a subproblem i, from the whole population.

$$\Gamma_i^{p\to c} = \max_{i=1,\ldots,k} \Gamma_{d.conv_y}^{p\to c}, \tag{6}$$

RDL is a diversity measurement quantity that indicates the amount of diversity loss of an individual solution between two consecutive generations. High values of RDL imply a reduction of the solution spread and this equation may indicated the amount of diversity loss.

This quantity is given by a division between the shortest distance of a parent, p, and offspring, c, to the line of convergence direction.

$$\Gamma_{d.conv_y}^{p\to c} = \frac{||p\prime - proj_{d.conv_y} p\prime||}{||c\prime - proj_{d.conv_y} c\prime||} \tag{7}$$

The numerator in 7 is the closest distance between the parent solution (p) to the convergence direction ($c_r - p_r$). While, the denominator in 7 is the closest distance between the offspring solution (c) to the convergence direction ($c_r - p_r$).

with $p\prime$ and $c\prime$ given by:

$$\begin{aligned} p\prime &= p - p_r, \\ c\prime &= c - c_s, \end{aligned} \tag{8}$$

with p_r and c_s being the parent and offspring objective vectors used to calculate the convergence direction in equation 11. That is the index s is equal to the index j used to calculate $conv_y$. The same principle is valid for the index r.

The vector projection between two vectors is defined as next.

$$proj_{d.conv_y} p\prime = \frac{d.conv_y \cdot p\prime}{|p\prime|^2} p\prime. \tag{9}$$

While the norm of $p\prime - proj_{d.conv_y} p\prime$ is calculated as follows.

$$||p\prime - proj_{d.conv_y} p\prime|| = sqrt(crossprod(proj_{d.conv_y} p\prime)) \tag{10}$$

and the norm of $c\prime - proj_{d.conv_y} c\prime$ is calculated similarly.

To estimate the convergence direction, $d.conv_y$, we need to have a offspring, c_j, that dominates a parent. Select a parent, p_h, solution that is closest to this offspring in the objective space.

For every weakly dominated parent, one convergence direction is calculated as in the next equation.

$$d.conv_y = c_j - p_h \tag{11}$$

The index j (for indexing offsprings, c_j) is selected from the set D_c. The index h is explained later.

$$D_c = \{d | \exists c_d \prec p_k, k \in 1, ..., N, d \in [1, ..., |C|]\}z \tag{12}$$

N is the parent population size, $|C|$ is the size of the offspring population C. In equation 12, the offspring c_d must weakly dominate at least one parent solution.

As from Zitzler et al. [34] study, weak dominance ($A \succeq B$) means that any solution in B is weakly dominated by a solution in A. However, this does not rule out equality, because $A \succeq A$ for all approximation sets $A \in \Omega$.

The index h (for indexing parents, p_h) comes from the following two equations.

$$h = \underset{k \in D_p}{argmin} ||p_k - c_j|| \tag{13}$$

$$D_p = \{k | \exists c_j \prec p_k, k \in 1, ..., N\} \tag{14}$$

D_p in equation 14 denotes the index set of parent solutions which are weakly dominated by c_j (the j index comes from equation 12).

IV. Experimental Design

This section introduces our experimental design. The question that we want to answer is how the proposed algorithm, MOEA/D-RAD, performs when compared with MOEA/D-DE and MOEA/D-GRA. This comparison is conducted by applying these three algorithms to the Black-Box Optimization Bi-Objective Benchmark (bbob-biobj) test suite [35].

To assess these comparisons, we use the hypervolume (HV) metric for measuring the quality of a set of obtained nondominated solutions found by the algorithms. Before calculating the HV value, the objective function was scaled using the ideal point and the nadir point of the all approximated PF of all algorithms, given a repetition. The reference point for the HV

Parameters	Values
Initial value u	0.5, for every subproblem
Population size	150
Neighborhood size T	20
δ_p	0.9
ϕ	0.5
η_m	20
p_m	0.03333333
n_r	2
Number of evaluations	60000
Number of repetitions	21

TABLE I
PARAMETER SETTINGS.

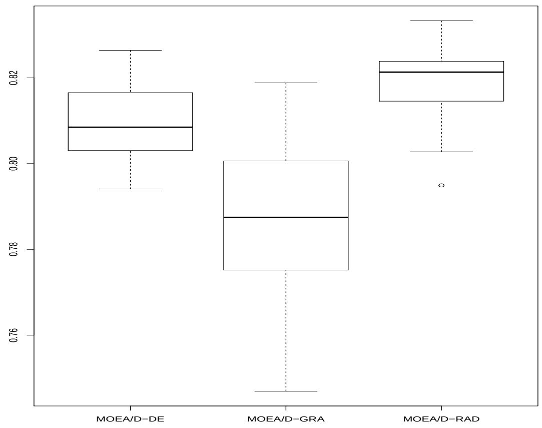

Fig. 1. Boxplot on the HV values of the 3 algorithms in the bbob-biobj-2.

calculation was set to $(1, 1)$. Higher values of the HV indicate better approximations.

In order to verify any statistical difference in the average performance given the different algorithms, the pairwise Wilcoxon nonparametric statistical test were used, with confidence interval $\alpha = 0.05$.

V. RESULTS

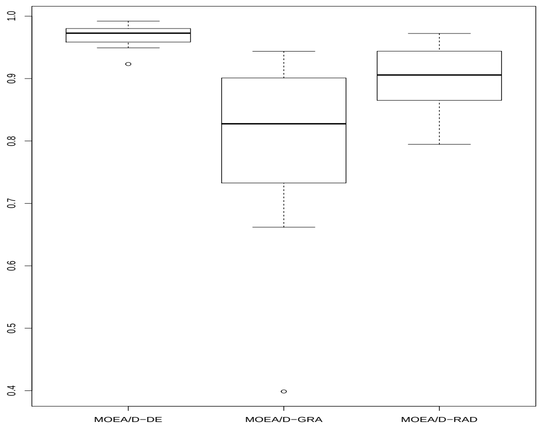

Fig. 2. Boxplot on the HV values of the 3 algorithms in the bbob-biobj-11.

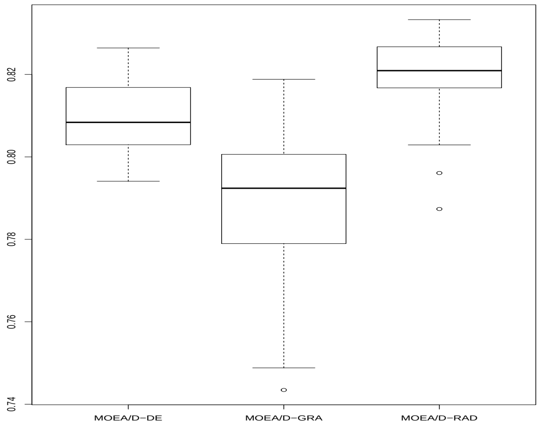

Fig. 3. Boxplot on the HV values of the 3 algorithms in the bbob-biobj-13.

REFERENCES

[1] K. Miettinen, "Nonlinear multiobjective optimization, volume 12 of international series in operations research and management science," 1999.

[2] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[4] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

[5] N. Beume, B. Naujoks, and M. Emmerich, "Sms-emoa: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[6] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 832–842.

[7] H. Li and Q. Zhang, "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.

[8] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[9] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 3, pp. 440–462, 2017.

[10] R. Hinterding, Z. Michalewicz, and A. E. Eiben, "Adaptation in evolutionary computation: A survey," in *Evolutionary Computation, 1997., IEEE International Conference on*. IEEE, 1997, pp. 65–69.

[11] K. De Jong, "Parameter setting in eas: a 30 year perspective," in *Parameter setting in evolutionary algorithms*. Springer, 2007, pp. 1–18.

[12] S. Meyer-Nieberg and H.-G. Beyer, "Self-adaptation in evolutionary algorithms," in *Parameter setting in evolutionary algorithms*. Springer, 2007, pp. 47–75.

[13] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of moea/d on cec09 unconstrained mop test instances," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*. IEEE, 2009, pp. 203–208.

[14] O. Kramer, "Evolutionary self-adaptation: a survey of operators and strategy parameters," *Evolutionary Intelligence*, vol. 3, no. 2, pp. 51–65, 2010.

Pairwise comparisons using Wilcoxon rank sum test		
\MOEA/D-RAD in function:	MOEA/D-DE	MOEA/D-GRA
bbob-biobj #2	0.00405 (>)	2e-07 (>)
bbob-biobj #11	3.1e-06 (<)	0.0026 (>)
bbob-biobj #13	0.00197 (>)	8.8e-07 (>)

[15] J. Zhang, W.-N. Chen, Z.-H. Zhan, W.-J. Yu, Y.-L. Li, N. Chen, and Q. Zhou, “A survey on algorithm adaptation in evolutionary computation,” *Frontiers of Electrical and Electronic Engineering*, vol. 7, no. 1, pp. 16–31, 2012.

[16] X. Cai, Y. Li, Z. Fan, and Q. Zhang, “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.

[17] A. Zhou and Q. Zhang, “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.

[18] Q. Kang, X. Song, M. Zhou, and L. Li, “A collaborative resource allocation strategy for decomposition-based multiobjective evolutionary algorithms,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018.

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

[20] V. Chankong and Y. Haimes, “Multiobjective decision making: Theory and methodology north holland,” *New York*, 1983.

[21] S. O. Hansson, “Decision theory a brief introduction. department of philosophy and the history of technology, royal institute of technology (kth) stockholm,” 2005.

[22] T.-C. Chiang and Y.-P. Lai, “Moea/d-ams: Improving moea/d by an adaptive mating selection mechanism,” in *Evolutionary Computation (CEC), 2011 IEEE Congress on*. IEEE, 2011, pp. 1473–1480.

[23] H.-L. Liu, F. Gu, and Q. Zhang, “Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems.” *IEEE Trans. Evolutionary Computation*, vol. 18, no. 3, pp. 450–455, 2014.

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

[25] Q. Zhang, A. Zhou, and Y. Jin, “Rm-meda: A regularity model-based multiobjective estimation of distribution algorithm,” *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 41–63, 2008.

[26] K. Deb, “Genetic algorithms in multimodal function optimization,” Ph.D. dissertation, Clearinghouse for Genetic Algorithms, Department of Engineering Mechanics, University of Alabama, 1989.

[27] J. Scott, “Fault tolerant design using single and multi-criteria genetic algorithms,” *Master’s thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology*, 1995.

[28] K. C. Tan, T. H. Lee, and E. F. Khor, “Evolutionary algorithms for multiobjective optimization: Performance assessments and comparisons,” *Artificial intelligence review*, vol. 17, no. 4, pp. 251–290, 2002.

[29] A. Farhang-Mehr and S. Azarm, “Diversity assessment of pareto optimal solution sets: an entropy approach,” in *Evolutionary Computation, 2002. CEC’02. Proceedings of the 2002 Congress on*, vol. 1. IEEE, 2002, pp. 723–728.

[30] K. Deb and S. Jain, “Running performance metrics for evolutionary multi-objective optimization,” 2002.

[31] K. Deb, M. Mohan, and S. Mishra, “A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions,” *KanGAL report*, vol. 2003002, pp. 1–18, 2003.

[32] S. Mostaghim and J. Teich, “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*. IEEE, 2003, pp. 26–33.

[33] K. C. Tan, C. K. Goh, A. Mamun, and E. Ei, “An evolutionary artificial immune system for multi-objective optimization,” *European Journal of Operational Research*, vol. 187, no. 2, pp. 371–392, 2008.

[34] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE Transactions on evolutionary computation*, vol. 7, no. 2, pp. 117–132, 2003.

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