On the Potential of Automated Algorithm Configuration on Multi-Modal Multi-Objective Optimization Problems

Jeroen Rook
Data Management and Biometrics
University of Twente, The Netherlands

Heike Trautmann
Statistics and Optimization, Univ. of Münster, Germany
University of Twente, The Netherlands

Jakob Bossek
Statistics and Optimization
University of Münster, Germany

Christian Grimme
Statistics and Optimization
University of Münster, Germany

ABSTRACT

Hardness of Multi-Objective (MO) continuous optimization problems results from an interplay of various problem characteristics, e. g. the degree of multi-modality. We present a benchmark study of classical and diversity focused optimizers on multi-modal MO problems based on automated algorithm configuration. We show the large effect of the latter and investigate the trade-off between convergence in objective space and diversity in decision space.

CCS CONCEPTS

• **Computing methodologies** → *Continuous space search.*

KEYWORDS

Multi-objective optimization, multi-modality, configuration

ACM Reference Format:
Jeroen Rook, Heike Trautmann, Jakob Bossek, and Christian Grimme. 2022. On the Potential of Automated Algorithm Configuration on Multi-Modal Multi-Objective Optimization Problems. In *Genetic and Evolutionary Computation Conference Companion (GECCO '22 Companion), July 9–13, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3520304.3528998

1 INTRODUCTION

In (evolutionary) MO optimization the predominant goal is to approximate the global set of trade-off solutions as good as possible w.r.t. convergence and diversity in objective space. Considering unconstrained *multi-objective problems* (MOP) [5], there may be multiple solutions in decision space with equal performance in objective space. These multi-modal MOPs impose specific challenges on algorithms due to their characteristics in terms of locally efficient sets, ridges and basin structures. Important is the distinction between *multi-global* and *multi-local* scenarios [8]: diverse solution sets in decision space map to the same images to the Pareto-front in objective space or, alternatively, the former might correspond to different local fronts in objective space. Of course, there might also be combinations of both. Classical evolutionary MO algorithms (EMOAs) such as NSGA-II, SMS-EMOA and MOEA/D have proven their performance with established parameter settings. Thus they can be used in an out-of-the-box manner for classical MOPs. For algorithms that address multi-modal MOPs there is not even an intuition of the achievable potential under a good configuration of their parameters, apart from a configuration study of MOEA/D on non-multi-modal MOPs [16]. This paper takes a further step in this direction and brings together multi-modal multi-objective evolutionary optimization with automated algorithm configuration (AAC) [12].

Here, we specifically investigate multi-global performance with the additional challenge that algorithms have to overcome the risk of getting stuck in local structures, possibly induced by a multi-local scenario. We show that MO algorithm ranking heavily depends on parametrization. Experiments further reveal a strong need for configuring for both maximum convergence in objective space as well as maximum solution diversity in decision space simultaneously in order to efficiently tackle multi-global optimization scenarios.

GECCO '22 Companion, July 9–13, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9268-6/22/07.
https://doi.org/10.1145/3520304.3528998

2 METHODOLOGY

Algorithms. Classical EMOAs concentrate on solution convergence and diversity in objective space. They all neglect diversity in decision space and thus may miss alternative global solutions of similar quality. Multi-modal MOPs have been tackled by integrating archiving, multiple populations, and niching techniques for preserving diverse solution sets in decision space [8]. Another stream of research exploit properties of landscape characteristics to move along local structures [9, 10, 14, 19, 23] and preserve locally efficient solutions [15]. Here, we pick representative approaches from each class: NSGA-II, SMS-EMOA, MOEA/D as classical EMOAs, plus Omni-Optimizer [7], MOLE [19] as an advanced implementation of MOGSA [10], and HIGA-MO [23].

Instances. We use a subset of well-known benchmark function collections: ZDT [25] consists of 6 bi-objective functions, each having a specific property that hampers the capability of EMOAs to address convergence and / or diversity. DTLZ [6] is an extension of ZDT towards decision and objective space dimension scalability. Within the MMF suite [24], the majority of problems is either unimodal or multi-global, but not multi-local. The bi-objective Black-Box Optimization Benchmark (BBOB) [22] contains 55 bi-objective test functions which originate from combinations of a subset of the SO BBOB benchmark [11] and show a strong level of multimodality.

Indicators. Researchers proposed many different indicators which often require a reference set [26]. In our setting we configure the algorithms to maximize either a) the *dominated hypervolume* (HV), i. e., a very prominent measure which calculates the space enclosed by the approximation set and an anti-optimal reference point or b) the *Solow-Polasky measure* which was designed to measure the amount of diversity between species in biology [21]. Both measures do not require reference sets. While the former is defined in objective space, the latter measures diversity in the decision space.

Automated algorithm configurators. Consider an algorithm A with a corresponding parameter configuration space Θ, which holds a list of parameters, respective domains and possible constraints on different parameter combinations. Given a set of problem instances I, the goal of AAC [12] is to find a optimal configuration $\theta^* \in \Theta$ that w. l. o. g. maximizes a quality metric Q on a problem set I, or formally defined as $\theta^* = \text{argmax}_{\theta \in \Theta}\ Q(A_\theta, I)$. In this work, the quality Q over I is computed as the mean performance over all instances I and an instance refers to a specific benchmark problem.

The number of distinct configurations in Θ is often very large and requires high-performing algorithm configurators. We will use SMAC [13], which is supported in Sparkle[1], and successfully demonstrated its performance in fields such as SAT and TSP (see [12] for details, also on alternative approaches such as irace or ParamILS).

3 EXPERIMENTS

A benchmark study is conducted for a variety of MO algorithms on a large set of, mainly multi-modal, MOPs of well-known benchmark sets. We investigate the extent of performance improvement induced by AAC w. r. t. both convergence in objective as well as diversity in decision space.

Experimental set-up. For all algorithms publicly available implementations [1, 3, 4, 18, 20, 23] were used. The R package `smoof` [2] was used to generate the problem instances. We compiled a set of 33 bi-objective problems from the 4 benchmark set (see Section 2) with 2-dimensional decision space. All problems from the ZDT, DTLZ and MMF benchmarks were selected, except ZDT5 and MMF13, since their decision space is not continuous or 2-dimensional. The remaining 3 instances are the highly multi-modal problems f_{46}, f_{47}, and f_{50} from bi-objective BBOB.

AAC experiments were conducted using SMAC with the Sparkle framework [2] Each configuration scenario consisted of 10 sequential configuration runs with each a limit of 250 algorithm calls and a wall clock time limit of 6 hours. Other configurator settings were set to their default. Each algorithm had a function call budget of 20 000. One exception to the latter is MOGSA because it has no termination criterion for budget. However, MOGSA often kept the number of function calls below the budget. The best configuration – based on the mean performance over all instances in the training set – of a scenario is picked for further validation on test instances.

Each configuration scenario maximized a single performance indicator (HV or SP). For HV calculation, in case of unknown reference points, we derived it from the maximum values of the non-dominated set out of the union of runs of all algorithms on the respective instance. During configuration, the HV metrics were normalized against the instances' maximum obtainable HV to assess the effect on the mean performance of each instance comparably. For SP the absolute scores were used, but to keep them aligned between algorithms and configurations, population sizes μ were set to 100. MOLE and MOGSA do not have populations and occasionally return much larger solution sets than they started with. In such cases, we reduced the solution sets by randomly sampling 2 000 points to keep the SP score computation possible.

Due to the limited number of instances we used leave-one-out cross validation for configuration. This implies that for each algorithm we ran 33 different configuration scenarios.

To validate the indicators' score, we ran each algorithm 25 times. Each of the 25 runs has a fixed random seed (e. g., each first run always had the same seed). The median over all runs was used to express each indicators' score. In the rare occasion that an algorithm could not find any non-dominated points that fell below the reference point, resulting in a HV of 0, we imputed them by the worst non-zero HV found in all other runs on that instance.

Configurability. To gauge how susceptible the algorithms are to improving their performance for one indicator specifically, we looked at both the indicator's performance after configuring for HV and after configuring for SP.

When configuring for HV, we observe a relative increase in HV comparing to default configuration for all algorithms and, unexpectedly, no relative decrease for SP, except for a small proportion of runs. In fact, for MOGSA, MOLE and HIGA-MO the SP even improves along with the HV. The amplitude of the relative improvement on HV differs widely between the algorithms. For example, NSGA-II, SMS-EMOA and Omni-Optimizer show little to none improvements. This can be attributed to their limited parameter spaces or, more likely, because their default parameters were already set to perform well for HV. In contrast, MOLE occasionally shows large improvements indicating that it can benefit from configured parameters significantly. The median value however is ≈ 0 , but MOLE also shows the strongest potential in increasing the SP-measure.

In case of configuring for SP, for all algorithms the SP increases, although sometimes at the cost of HV. The latter is most dominant for MOEA/D and Omni-Optimizer. This suggests that these algorithms might be over-fitted for optimizing HV. In general, we observe considerable variability across repeated runs for HIGA-MO, MOGSA, and MOLE, which exploit local structures.

The resulting parameter configurations strengthen the aforementioned findings. There is a tendency that parameters configured for SP tend to differ substantially more from the default settings compared to parameter settings optimized for HV. For MOLE, however, parameters for both configurations largely differ from the default parameters. This indicates that these default parameters are likely not targeted to perform well on the selected indicators and should be adjusted in order to exhibit good convergence properties.

Competitiveness. We complement the perspective of configurability by an investigation of the competitiveness of the considered algorithms before and after configuration. In that context, we focus on six scenarios: two rankings w. r. t. HV and SP of unconfigured algorithms as well as the rankings w. r. t. both HV and SP, when the algorithms are either configured to maximize HV or SP, respectively.

[1] Accessible through `ada.liacs.nl/projects/sparkle`
[2] All code is available at: `github.com/jeroenrook/MMMOO-AAC-experiments`

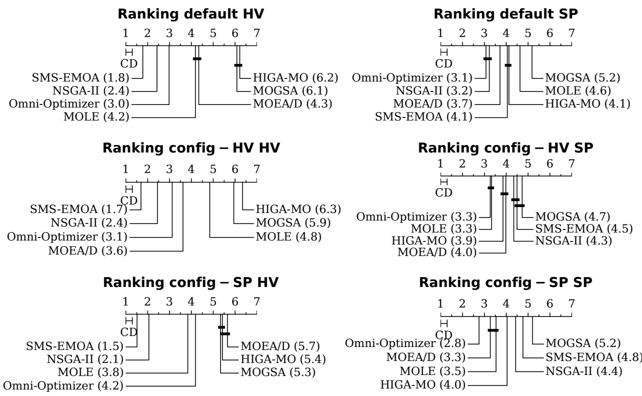

Figure 1: Critical distance plots over the average rank of an algorithm over all individual runs (seed-instance). The figures on the left show the ranking of HV and the figures on the right for SP. The rows are the ranks for default, configured for HV and configured for SP, respectively.

The scenarios are denoted `default HV`, `default SP`, `config-HV HV`, `config-HV SP`, `config-SP HV`, and `config-SP SP`, where the first component denotes the configuration objective and the second component denotes the ranking objective. The `default` scenarios use standard (recommended) parameters for the algorithms.

We ranked the algorithms for each distinct seed-instance pair from which we computed the average rank for each algorithm. A Nemenyi test [17] with ($\alpha < 0.1$) was conducted to determine the critical distance (CD) of 0.15 between ranks to be significantly better or worse than the other algorithms (see Figure 1).

In these rankings we see that SMS-EMOA and NSGA-II are in all cases leading, when HV is considered as ranking objective. In terms of SP, both algorithms are partly outranked by Omni-Optimizer and MOEA/D for the default scenario and become even worse after configuration. Omni-Optimizer shows strong performance in all the SP rankings but also MOLE performs much better. MOLE even shares rank 1 with Omni-Optimizer in the `config-HV SP` scenario. MOGSA is consistently outranked and HIGA-MO shows a similar pattern, except for SP where it ranks better than SMS-EMOA and NSGA-II when the algorithms were configured for SP.

Relative indicator loss. We now focus on the indicators' performance differences between both configurations to investigate the potential of obtaining maximum convergence in objective space as well as maximum solution diversity in decision space. Specifically, we looked at the relative change between the non-configured indicator's performance and its performance when it was configured for that indicator, i. e., the relative change of HV after configuring for SP and after configuring for HV. A positive relative change indicates a *loss* in what potentially can be achieved for that indicator. Hence, we refer to the relative changes as losses.

Figure 2 shows the relative loss on both indicators for each algorithm. The losses on HV (blue boxplots), if configured for SP, show that all algorithms suffer some degree of relative losses ranging from the largest losses for MOEA/D and Omni-Optimizer and the smallest for HIGA-MO, NSGA-II and SMS-EMOA.

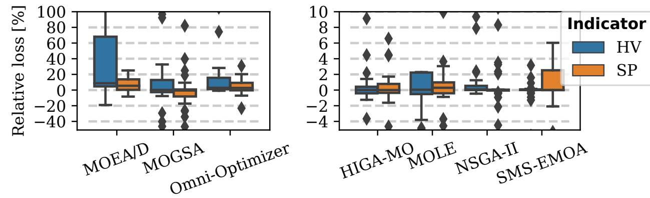

Figure 2: Relative loss between the configurations for HV and SP on the indicators that were not targeted in the configurations, respectively.

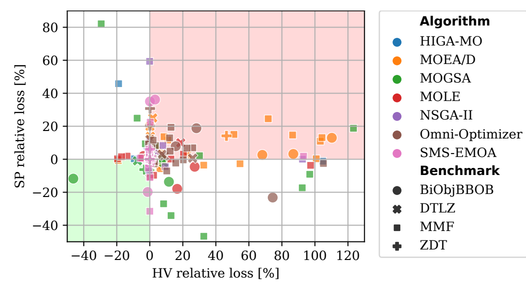

Figure 3: The relative loss for each indicator when it was configured for the opposite indicator. Each points represents the relative loss of an algorithm-instance pair.

For SP (orange boxplots), i. e., configured on HV, we see that all algorithms, except MOGSA, suffer a loss on their potential as well. For MOGSA this means that there is a relative gain, i. e., the SP performance was better when configuring for HV. Again, the amplitude in the relative loss varies across the tested algorithms. MOEA/D and Omni-Optimizer show the largest loss here and are the only losses that are significant[3] for SP. Overall, we notice that the losses for HV are higher than the losses for SP.

Figure 3 combines the losses for each indicator and presents it in a scatter plot. Here, each point represents the loss for HV and SP for one of the 231 algorithm-instance pairs. These pairs are divided into three partitions; a green partition holding 30 (13%) pairs that show no loss on both indicators, a white partition holding 109 (47%) pairs have a loss on only one indicator and a red partition with the remaining 92 (40%) pairs that have a loss on both indicators. This shows that for the majority of pairs we are not achieving their true potential for both indicators simultaneously. The proportion (adjusted for their total proportion) of benchmarks is consistent across each partition. This indicates that the benchmarks sets are not particularly more sensitive or insensitive to these losses. The proportion of presence of the algorithms for each partition shows that MOEA/D and Omni-Optimizer are above average (27% and 23% resp. over the average of 14%) in the partition with losses on both indicators. Omni-Optimizer is also the only algorithm that never displays improvements for both indicators.

Discussion. From the presented experimental results, we derive several insights: (1) Compared to the other considered approaches SMS-EMOA and NSGA-II often excel in the task they are designed

[3]based on a Wilcoxon signed-rank test with $\alpha = 0.1$ of the median indicator scores.

for: they create a good approximation of the Pareto front. Combined with the observation of low configurability (see Section 3), we cannot expect a large influence of our configuration on these overall results. (2) Conversely, this also implies that configuration w. r. t. SP has little effect on these algorithms' performance. As expected, it also shows, that the objective-space-focused design of SMS-EMOA and NSGA-II holds no potential w. r. t. conserving alternative solutions in decision space which results in a loss of diversity. (3) For algorithms like Omni-Optimizer and MOLE the configurations hold the potential to improve their performance compared to other established methods over all considered benchmarks. Remember, that specifically MOLE is a local optimizer. Still, it is able to compete with global approaches. Its approach of traversing multiple local (and global) efficient sets during directed MO descent certainly helps in preserving alternative solutions. (4) Overall, the potential in configuration w. r. t. multiple goals like convergence and diversity, is large for methods which either exploit local structures (like MOLE) or at least preserve alternative solutions in archives, niches, or multiple populations (like e. g. Omni-Optimizer or MOEA/D). Future research will show, whether this finding also generalizes to related approaches. (5) The results show that MOLE which is conceptually similar to the original MOGSA [10] approach but more developed, is competitive with other optimizers. (6) We observe that when configuring for either HV or SP, in most cases we have to hazard the consequence of loosing performance regarding the other criterion. So there is a strong need for perspectively focusing on reaching the optimal trade-off between both criteria by means of multi-objective AAC.

4 SUMMARY AND CONCLUSION

We demonstrate the large potential of automated algorithm configuration in the evolutionary MO domain focusing on multi-modal problems for which both decision space diversity as well as convergence in objective space is of crucial interest. Specialized algorithms, which exploit local structures or include archiving and niching techniques are contrasted to more general, commonly used state-of-the-art MO algorithms such as NSGA-II and SMS-EMOA. We show that suitably set parameters in most cases substantially improve algorithm performance, especially w. r. t. decision space diversity. From our experiments, however, we explicitly derive the necessity of taking a multi-objective perspective in automated algorithm configuration by simultaneously optimizing for convergence and decision space diversity.

REFERENCES

[1] Jakob Bossek. 2017. Ecr 2.0: A Modular Framework for Evolutionary Computation in R. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO '18)*. ACM, Berlin, Germany, 1187–1193. https://doi.org/10.1145/3067695.3082470

[2] Jakob Bossek. 2017. smoof: Single- and Multi-Objective Optimization Test Functions. *The R Journal* 9, 1 (2017), 103–113. https://doi.org/10.32614/RJ-2017-004

[3] Jakob Bossek and Kalyanmoy Deb. 2021. *omnioptr: Omni-Optimizer*. https://github.com/jakobbossek/omnioptr R package version 1.0.0.

[4] Felipe Campelo, Lucas S. Batista, and Claus Aranha. 2020. The MOEADr Package: A Component-Based Framework for Multiobjective Evolutionary Algorithms Based on Decomposition. *Journal of Statistical Software* 92, 6 (2020), 1–39. https://doi.org/10.18637/jss.v092.i06

[5] Carlos Artemio Coello Coello, David A. van Veldhuizen, and Gary B. Lamont. 2007. *Evolutionary Algorithms for Solving Multi-Objective Problems* (2nd ed.). Springer.

[6] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. 2005. *Scalable Test Problems for Evolutionary Multiobjective Optimization*. Springer London, London, 105–145. https://doi.org/10.1007/1-84628-137-7_6

[7] Kalyanmoy Deb and Santosh Tiwari. 2005. Omni-optimizer: A Procedure for Single and Multi-objective Optimization. In *Evolutionary Multi-Criterion Optimization*. Springer Berlin Heidelberg, Berlin, Heidelberg, 47–61.

[8] Christian Grimme, Pascal Kerschke, Pelin Aspar, Heike Trautmann, Mike Preuss, Andre H. Deutz, Hao Wang, and Michael Emmerich. 2021. Peeking beyond peaks: Challenges and research potentials of continuous multimodal multi-objective optimization. *Computers & Operations Research* 136 (2021), 105489. https://doi.org/10.1016/j.cor.2021.105489 Publication status: Published.

[9] Christian Grimme, Pascal Kerschke, Michael T. M. Emmerich, Mike Preuss, André H. Deutz, and Heike Trautmann. 2019. Sliding to the Global Optimum: How to Benefit from Non-Global Optima in Multimodal Multi-Objective Optimization. In *AIP Conference Proceedings* (Leiden, The Netherlands). AIP Publishing, 020052–1–020052–4.

[10] Christian Grimme, Pascal Kerschke, and Heike Trautmann. 2019. Multimodality in Multi-Objective Optimization — More Boon than Bane?. In *Proceedings of the 10^{th} International Conference on Evolutionary Multi-Criterion Optimization (EMO)* (East Lansing, MI, USA) *(Lecture Notes in Computer Science (LNCS), Vol. 11411)*, Kalyanmoy Deb, Erik Goodman, Carlos A. Coello Coello, Kathrin Klamroth, Kaisa Miettinen, Sanaz Mostaghim, and Patrick Reed (Eds.). Springer, 126 – 138. https://doi.org/10.1007/978-3-030-12598-1_11

[11] Nikolaus. Hansen, Steffen Finck, Raymond Ros, and Anne Auger. 2009. *Real-Parameter Black-Box Optimization Benchmarking 2009: Noiseless Functions Definitions*. Technical Report RR-6829. INRIA. http://hal.inria.fr/inria-00362633/en/

[12] Holger H. Hoos. 2012. Automated Algorithm Configuration and Parameter Tuning. In *Autonomous Search*, Youssef Hamadi, Eric Monfroy, and Frédéric Saubion (Eds.). Springer, 37–71. https://doi.org/10.1007/978-3-642-21434-9_3

[13] Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. 2011. Sequential Model-Based Optimization for General Algorithm Configuration. In *Proceedings of the 5th International Conference on Learning and Intelligent Optimization (LION 5)*. Springer Berlin Heidelberg, 507–523.

[14] Pascal Kerschke and Christian Grimme. 2017. An Expedition to Multimodal Multi-Objective Optimization Landscapes. In *Proceedings of the 9^{th} International Conference on Evolutionary Multi-Criterion Optimization (EMO) (Lecture Notes in Computer Science (LNCS), Vol. 11411)*, Heike Trautmann, Günter Rudolph, Kathrin Klamroth, Oliver Schütze, Margaret Wiecek, Yaochu Jin, and Christian Grimme (Eds.). Springer, Münster, Germany, 329 – 343. https://doi.org/10.1007/978-3-319-54157-0_23

[15] Yiping Liu, Hisao Ishibuchi, Yusuke Nojima, Naoki Masuyama, and Yuyan Han. 2019. Searching for Local Pareto Optimal Solutions: A Case Study on Polygon-Based Problems. In *2019 IEEE Congress on Evolutionary Computation (CEC)* (Wellington, New Zealand). IEEE, 896 – 903.

[16] Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Mauro Birattari, and Thomas Stützle. 2016. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives* 3 (2016), 43–58. https://doi.org/10.1016/j.orp.2016.09.002

[17] Peter Bjorn Nemenyi. 1963. *Distribution-free multiple comparisons*. Princeton University.

[18] person). 2019. *mogsa: A Multi-Objective Optimization Algorithm Based on Multi-Objective Gradients*. https://github.com/kerschke/mogsa R package version 1.0.

[19] Lennart Schaepermeier. 2021. *Multimodal Search Structures in Continuous Multi-Objective Optimization*. Master's thesis. University of Münster.

[20] Lennart Schäpermeier. 2022. *An R Package Implementing the Multi-Objective Landscape Explorer (MOLE)*. https://github.com/schaepermeier/moleopt

[21] Andrew R. Solow and Stephen Polasky. 2006. Measuring biological diversity. *Environmental and Ecological Statistics* 1 (2006), 95–103.

[22] Tea Tušar, Dimo Brockhoff, Nikolaus Hansen, and Anne Auger. 2016. COCO: The Bi-Objective Black Box Optimization Benchmarking (bbob-biobj) Test Suite. *arXiv preprint* abs/1604.00359 (2016).

[23] Hao Wang, André Deutz, Thomas Bäck, and Michael Emmerich. 2017. Hypervolume Indicator Gradient Ascent Multi-Objective Optimization. In *9th International Conference on Evolutionary Multi-Criterion Optimization - Volume 10173* (Münster, Germany) *(EMO 2017)*. Springer-Verlag, Berlin, Heidelberg, 654–669. https://doi.org/10.1007/978-3-319-54157-0_44

[24] Caitong Yue, Boyang Qu, Kunjie Yu, Jing Liang, and Xiaodong Li. 2019. A Novel Scalable Test Problem Suite for Multimodal Multiobjective Optimization. *Swarm and Evolutionary Computation* 48 (2019), 62 – 71.

[25] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. 2000. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation (ECJ)* 8, 2 (2000), 173 – 195.

[26] Eckart. Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane G. da Fonseca. 2003. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation (TEVC)* 7, 2 (2003), 117–132. https://doi.org/10.1109/TEVC.2003.810758