Archiver Effects on the Performance of State-of-the-art Multi- and Many-objective Evolutionary Algorithms

Leonardo C. T. Bezerra
Instituto Metrópole Digital,
Universidade Federal do Rio Grande do Norte
Natal, RN, Brazil
leobezerra@imd.ufrn.br

Manuel López-Ibáñez
Alliance Manchester Business School
University of Manchester
Manchester, UK
manuel.lopez-ibanez@manchester.ac.uk

Thomas Stützle
IRIDIA, CoDE
Université Libre de Bruxelles
Brussels, Belgium
stuetzle@ulb.ac.be

ABSTRACT

Early works on external solution archiving have pointed out the benefits of unbounded archivers and there have been great advances, theoretical and algorithmic, in bounded archiving methods. Moreover, recent work has shown that the populations of most multi- and many-objective evolutionary algorithms (MOEAs) lack the properties that one would desire when trying to find a bounded Pareto-optimal front. Despite all these results, many recent MOEAs are still being proposed, analyzed and compared without considering any kind of archiver assuming their additional computational cost is not justified. In this paper, we investigate the effect of using various kinds of archivers, improving over previous studies in several aspects: (i) the parameters of MOEAs with and without an external archiver are tuned separately using automatic configuration methods; (ii) we consider a comprehensive range of problem scenarios (number of objectives, function evaluations, computation time limit); (iii) we employ multiple, complementary quality metrics; and (iv) we study the effect of unbounded archivers and two state-of-the-art bounded archiving methods. Our results show that both unbounded and bounded archivers are beneficial even for many-objective problems. We conclude that future proposals and comparisons of MOEAs must include archiving as an algorithmic component.

CCS CONCEPTS

• Applied computing → Multi-criterion optimization and decision-making; **• Theory of computation → Bio-inspired optimization**;

KEYWORDS

Multi-objective optimization, evolutionary algorithms, algorithm configuration, experimental analysis, archiving

GECCO '19, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6111-8/19/07…$15.00
https://doi.org/10.1145/3321707.3321789

ACM Reference Format:
Leonardo C. T. Bezerra, Manuel López-Ibáñez, and Thomas Stützle. 2019. Archiver Effects on the Performance of State-of-the-art Multi- and Many-objective Evolutionary Algorithms. In *Genetic and Evolutionary Computation Conference (GECCO '19), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3321707.3321789

1 INTRODUCTION

Multi-objective optimization (MO) has become one of the most prominent applications of evolutionary algorithms (EAs). Indeed, the research on multi-objective evolutionary algorithms (MOEAs) has produced over the past three decades a rich literature on algorithms and theoretical foundations that have been instrumental for advancing the broader MO field [9, 10]. However, until recently it remained unclear what actually is the state-of-the-art in MOEAs [4] for two major reasons. The first is the speed at which novel algorithms are proposed, posing a significant challenge for benchmarking efforts. The second is the lack of rigor in the assessment of these more recently proposed algorithms, best illustrated by several shortcomings in many comparisons. A first shortcoming is the limited number of algorithms usually being compared, either including only "popular" MOEAs, without citing any independent experimental comparison supporting this choice, or excluding other MOEAs due to misconceptions about their performance or suitability for specific scenarios. For instance, there is a widespread belief that SMS-EMOA [3] is too computationally expensive for more than three objectives. However, thanks to recent advances in the computation of the hypervolume [2, 8, 14, 31], i.e., novel algorithms that scale better with respect to the number of objectives, SMS-EMOA is able to outperform other MOEAs specifically designed for many-objective problems, so long as it is properly tuned for this scenario [6]. In a similar way, algorithms like NSGA-II [12] and SPEA2 [35] are regarded unsuitable for many-objective optimization. Yet, the major issue with these algorithms concerns environmental selection [25], so external archiving is likely to improve their performance.

Several other shortcomings can also be identified, such as (i) comparing algorithms that are based on different underlying EAs, thus potentially mixing the effects of the MO and the EA components; (ii) the lack of proper algorithm configuration, generally reusing familiar parameter settings or settings based on limited and often unstructured preliminary experiments; (ii) using a single performance metric to evaluate results, even when it is widely known that the most relevant metrics may disagree [5, 20]; and/or (iv) overgeneralizing results when the stopping criteria and/or problem sizes are not broad enough.

In recent work [6], we have addressed these issues in a first effort to assess the state-of-the-art in MOEAs. In more detail, we have compared 9 MOEAs and their variants (totalizing 14 independently proposed algorithms) using the most relevant artificial benchmark sets, underlying EAs, performance metrics, and algorithm configuration tools available. In addition, that analysis has considered different numbers of objectives and stopping criteria, increasing the soundness and generality of the conclusions. Some of the insights identified match the common sense in the community, such as the effect of dominance resistance and the disagreement between performance metrics. Others made evident that some conclusions from prior, less comprehensive studies may have been premature, such as the improvements of more recent algorithms over older ones, which have been over-estimated.

This paper continues the effort to assess the state-of-the-art in MOEAs, specifically for what concerns the effects of archivers on their performance. Our work is closely related to the work of Tanabe et al. [28], who compared a number of MOEAs using a wide range of numbers of objectives and stopping criteria, with and without archivers. Unfortunately, in that work the parameters of the algorithms were not configured to perform at their best, with no choice of underlying EAs, and quality was evaluated in terms of a non-standard metric. In this paper, we overcome these limitations by applying automatic algorithm configuration before comparing algorithms and by using standard quality metrics to ensure the generality of our conclusions.

In the first part of our work, we use MOEAs with unbounded, external archivers, and automatically configure them to adapt to this component. Results are strongly affected by all the experimental factors we consider, i.e., the factors considered in [6] plus archive type, truncation strategy, and capacity. Results show consistent improvements in the performance of all MOEAs on all scenarios. Surprisingly, archiver benefits are even stronger on scenarios where its overhead is expected to be too high for practical purposes, namely many-objective scenarios with constrained runtime. In the second part of our work, we study the effect of bounded archivers and the loss in quality for various archiving methods and bound sizes. Results show losses when compared to unbounded archivers, which, however, are in general acceptable, depending on the capacity of the archive. Our results suggest that a promising MOEA design combines smaller population sizes, which increase convergence pressure and reduce search overhead, with bounded archiving methods that ensure high-quality final approximation fronts.

Although our work produces the relevant insights discussed above, the data generated during our research may also be examined with respect to other research questions. We therefore add the produced data to our public MOEA experimental data repository[1] to stir further investigations.

The remainder of this paper is structured as follows. Section 2 briefly reviews the MOEA literature, highlighting the algorithms we consider and discussing the related literature on archiving. In Section 3, we revisit the previously proposed assessment setup from [6], which we reuse in this work for comparability. Sections 4–5 respectively detail how we extend it to account for the effects of unbounded and bounded archivers, and discusses the major insights from these investigations. We conclude in Section 6, highlighting promising possibilities for future work.

[1]https://github.com/leobezerra/moea-benchmark

2 BACKGROUND

Our experimental investigation includes a number of relevant MOEAs representative of the literature. In this section, we first briefly review the MOEAs we assess, explaining our motivation for this selection. Next, we discuss archiving methods, highlighting the ones we adopt in this investigation.

2.1 Multi-objective evolutionary algorithms

As previously discussed, the more than three decades of MOEA research has produced a number of algorithms. Here, we consider algorithms classified into three main groups:

Dominance-based MOEAs were the first proposed with the goal of demonstrating that it was possible to achieve a set of mutually nondominated solutions from a single run. This was achieved through the combination of a fitness and a diversity component, respectively responsible for convergence and keeping a spread, well-distributed approximation front. As a historical baseline,[2] we consider **MOGA** [16] from this earliest group of MOEAs. Later, it became default practice for dominance-based algorithms to adopt elitism; these MOEAs are represented in our assessment by **NSGA-II** [12] and **SPEA2** [35].

Indicator-based MOEAs were proposed as an improvement over dominance-based ones. They adopt quality indicators as a refinement of dominance-based fitness components, given that indicators are able to discriminate between nondominated solutions and are expected to be less affected by the increase in the number of objectives. We take **IBEA** [34], **SMS** [3], and **MO-CMA-ES** [18, 30] as examples of early MOEAs from this group. Later, indicator-based MOEAs were proposed to address *many-objective optimization*, i.e., problems with more than three objectives. We take **HypE** [1] to represent these many-objective indicator-based MOEAs.

Decomposition-based proposals were among the first in the MOEA literature, yet did not gain as much attention as dominance-based ones. In part, this is explained by the geometries of artificial problems typically employed in the assessment of MOEAs, which may deceive decomposition-based algorithms into searching along non-promising directions of the objective space. To avoid some of such potential disadvantages, **MOEA/D** [32, 33], proposed a dynamic selection of search directions. A number of decomposition-based algorithms were more recently proposed for many-objective optimization, from which we select **NSGA-III** [11].

2.2 Unbounded and Bounded Archivers

Archiving is the strategy of storing a (external) set of nondominated solutions, an *archive*, in addition to the MOEA main population [29]. In their most usual form, archives are external to the evolutionary

[2]We choose this approach over adopting a random search or dummy optimizer for baseline, because adopting MOGA as a baseline indicates the actual improvements achieved by MOEA research over the years.

process, that is, they do not influence the trajectory of the search, and keep only nondominated solutions. In this paper, we use the term *archiver* to denote the algorithm that decides which solutions are stored in an archive.

There are two major motivations for employing archiving, related to the overhead and progress of the evolutionary process. Concerning overhead, increasing the number of objectives naturally pushes approximation fronts to present more solutions, yet large population sizes incur an increased computational overhead. This large size is also detrimental for the progress of evolutionary process, since it becomes more difficult to select solutions for variation. Progress is also affected by the quality of solutions maintained: keeping only nondominated solutions in a population would mean a high risk of stagnation or converging to a narrow region of the objective space. Since the population size needs to be kept constrained, this effectively means nondominated solutions get eventually discarded, sometimes in exchange for dominated ones. In the long run, this may result in the population not converging to optimality (or at all), and that solutions (or even the whole population) may be dominated by solutions (or populations) from a previous stage of the run [25]. The use of an external archive alleviates these problems by freeing the population being evolved from having to maintain the best possible approximation front found so far.

Ideally, archives would be **unbounded** in order to store every nondominated solution found and fully approximate the whole Pareto-optimal front. Yet, it is common to bound archives through some truncation technique for practical reasons. Bounded archives may suffer from the same issues just discussed for populations. We also differentiate *offline* and *online* archiving. In offline archiving, an unbounded archive is used during the run and truncated only when the run is over; in online archiving, a bounded archive is used all along the run.[3]

Below, we briefly summarize the most important properties archives should present [25]:

Monotonicity means the archive does not *deteriorate*, i.e., a solution at a given stage of the run is never dominated by a solution previously discarded.

$\triangleleft$-monotonicity means the archive does not $\triangleleft$-*deteriorate*, i.e., the approximation front at a later stage of the run is never worse, according to the weakly dominance relation among fronts [36], than a front at an earlier stage.

Limit-stability means the archive eventually converges to an approximation front that does not change.

Limit-optimality means the archive converges to a subset of the Pareto-optimal front.

So far, no archiver (except an unbounded one) has been proven to maintain archives that present all of the above desirable properties. Both multi-grid (**MGA** [23]) and hypervolume (**AA$_s$** [21]) archivers have demonstrated all but monotonicity, and for this reason we adopt them in our assessment. AA$_s$ discards solutions that contribute the least to the hypervolume of the archiver. MGA dynamically creates a hierarchical grid of boxes that discretize the objective space, and discards the solution that is dominated by another box at some level of the hierarchy, moving up the hierarchy as necessary in order to maintain the archive size.

[3]One could further differentiate between adding/discarding a single (sequential) or multiple (one shot) solutions at a time. We adopt the sequential approach and refrain from further investigating the effects of such distinction to keep our assessment feasible, since it already represents 10 CPU-years effort.

Table 1: Experimental setup adopted from [6]. The differences in setup are underlined.

Factor	Details
Problems	DTLZ{2, 4–7}; WFG1–9
M	{2, 3, 5, 10}
n_{var}	{20, 21, . . . , 60}
$FE_{\max}$	{2 500, 10 000, 40 000}
n_{testing}	{30, 40, 50}
n_{tuning}	$n_{\text{var}} \setminus n_{\text{testing}}$
$t_{\max}$	1h ($FE_{\max}$ = 2 500) 10min (otherwise)
Testing	I_H^{rd}, $I_{\epsilon+}$, I_{IGD}, $\underline{I_{IGD+}}$
	25 repetitions

Factor	Details
u	$\begin{cases} [10]^M, & \text{if } M \in \{2,3\} \\ [15]^5, & \text{if } M = 5 \\ [25]^{10}, & \text{if } M = 10 \end{cases}$
r	$1.1 \cdot \mathbf{u}$
Config.	irace
	20 000 MOEA runs
	$I_{\epsilon+}$ (M = 10)
	I_H^{rd} (otherwise)

As an implementation note, we run all MOEAs with an unbounded archiver, that is, discarding only dominated solutions, and the final archive is returned once the run terminates. Then, we apply the bounded archivers (MGA and AA$_s$) to the sequence of solutions found in the archive. We adopt this approach to save computational effort and make the study feasible, since the bounded archivers take non-negligible computation times, specially in many-objective scenarios. Arguably, just storing all solutions ever found and discarding dominated only at the end of the run could be more computationally efficient, specially as the number of objectives increases. Yet, as we will later discuss, results demonstrate that online unbounded archiving still leads to improved results with respect to not using archiving, specially for many-objective optimization problems.

In the next section, we detail the experimental setup we adopt to investigate the effects from these archivers on the performance of the state-of-the-art MOEAs.

3 EXPERIMENTAL SETUP

To ensure the generality of our conclusions, we follow the setup proposed in our recent assessment of the state-of-the-art in MOEAs [6], briefly summarized in Table 1. In this section, we first review that setup, and in the next sections we explain how we extend it to assess the effects of archiving. Several experimental factors are considered, which are briefly explained below. For further details on any factor, we refer to the original paper.

Artificial benchmark sets: We consider the DTLZ [13] and WFG [17] box-constrained continuous problem sets, with varying numbers of objectives ($M \in \{2, 3, 5, 10\}$) and variables ($n_{\text{var}} \in \{20, 21, \ldots, 60\}$). Problems DTLZ1 and DTLZ3 are left out of the investigation since MOEAs are unable to tackle them when their number of variables is scaled, leading to ceiling effects [6].

Stopping criteria: We simulate different computational costs of function (or solution) evaluations (FE) by providing MOEAs with varying budgets $FE_{\max} \in \{2\,500, 10\,000, 40\,000\}$. To ensure the feasibility of our investigation, algorithms are terminated if they exceed a maximum runtime $t_{\max} = 1$ hour ($FE_{\max} = 2\,500$) or $t_{\max} = 10$ minutes (otherwise). The maximum runtime is longer when $FE_{\max} = 2\,500$ since in practice such a reduced FE budget would only be adopted in the case of computationally costly FEs, thus requiring a longer cutoff time.

Performance metrics: We try to evaluate desirable features of approximation fronts, in particular convergence (related to closeness to optimality of individual solutions), spread (related to the extent of the approximation front), and distribution (related to the evenness of the approximation front) [20]. The original setup adopted the relative deviation from an approximation of the optimal hypervolume I_H^{rd}, the unary additive ϵ-metric $I_{\epsilon+}$, and the inverted generational distance I_{IGD} performance metrics to assess these features. Here, we additionally consider the I_{IGD+} metric [19], as it is a Pareto-compliant variant of the I_{IGD}. Since MOEAs are stochatic, results presented have been collected from 25 independent runs with pre-selected seeds. Before computing metrics, fronts are preprocessed by discarding outliers dominated by point $\mathbf{u}$, which is also used to compute the reference point for I_H^{rd}.

Underlying evolutionary algorithms: We consider a separation between MO and EA components. This way, the same set of MO components may be used in combination with different underlying EAs, namely *genetic algorithms*[4] and *differential evolution*. As a result, algorithms like NSGA-II and DEMO [26], which differ only in their underlying EA, are both referred to as NSGA-II (the firstly proposed one). The choice of which to use as the underlying EA is offered to all algorithms as a parameter to be configured, except for MO-CMA-ES [18, 30], which uses CMA-ES as the underlying EA.

Algorithm configuration: Besides the underlying EA choice, MOEAs generally present numerical parameters to be configured. We use irace [24] to automatically configure these parameters for each MOEA on each scenario (that is, each setting of the number of objectives and the maximum number of function evaluations) in order to compare them at their best performance. The irace procedure is guided by I_H^{rd}, when $M < 10$ and by $I_{\epsilon+}$, otherwise. Each irace run is allowed 20 000 MOEA runs. To ensure the separation between tuning and testing benchmark sets, we reserve number of variables $n_{\text{testing}} \in \{30, 40, 50\}$ for testing and each run of irace considers as training set all benchmark functions with a common number of objectives (given by the scenario) and $n_{\text{var}} \setminus n_{\text{testing}}$.

In the next sections, we describe the experiments we conducted to respectively assess the effects of unbounded and bounded archiving on the performance of the selected MOEAs. Note that for each setting and whether using or not archivers, the algorithms have undergone the same automatic configuration process using irace.

4 UNBOUNDED ARCHIVING

In this first part of the investigation, we assess MOEAs coupled with an unbounded archive. Instead of tacitly assuming that the addition of an unbounded archive does not interact with any MOEA parameter, we run irace to tune the search behavior of the MOEAs to the presence of this new component. For brevity, these tuned settings are provided as supplementary material [7], but the most important change is that the population sizes become much smaller

[4]For the purposes of our analysis, we include evolution strategies within the term genetic algorithms.

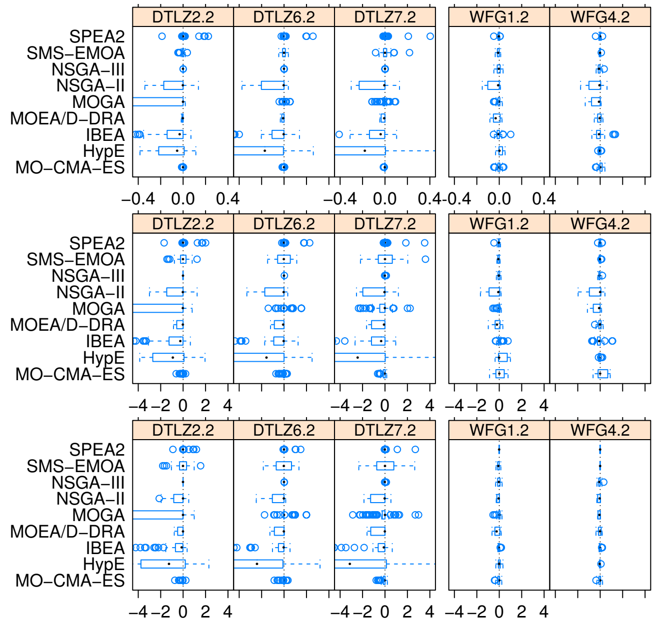

Figure 1: Performance differences between runs from MOEAs with and without unbounded archiving on the scenario with $M = 2$ and $FE_{\max} = 10\,000$ for selected problems with $n_{\text{var}} = 40$. From top to bottom, I_H^{rd}, $I_{\epsilon+}$, and I_{IGD}.

than for the MOEAs that are tuned without an unbounded archive. Effectively, the addition of the unbounded archive allows the population sizes to decrease without affecting the size of the final set returned while at the same time increasing convergence pressure and making mating selection more effective. The only exception is MOEA/D, for which population sizes change little. Other changes are punctual and algorithm-specific, and do not follow a clear-cut pattern.

In the remainder of this section, we first investigate the differences in performance incurred by the adoption of unbounded archiving. Later, we perform an overall comparison between MOEAs with unbounded archives to understand how this component affects the state-of-the-art for multi- and many-objective continuous optimization.

4.1 Differences in performance from unbounded archiving

Since unbounded archives present a trade-off between the quality of the approximation front produced and the added computational overhead, we initially assess which MOEAs (and on which scenarios they) actually benefit from this component. We start this analysis considering scenarios with $FE_{\max} = 10\,000$. Figures 1 and 2 respectively illustrate performance differences according to I_H^{rd} (top), $I_{\epsilon+}$ (middle), and I_{IGD} (bottom) when $M = 2$ and $M = 10$ on selected problems (and sizes). Differences are calculated by pairing runs with and without unbounded archiving, and negative values represent improvements provided by the unbounded archiver. I_{IGD+}

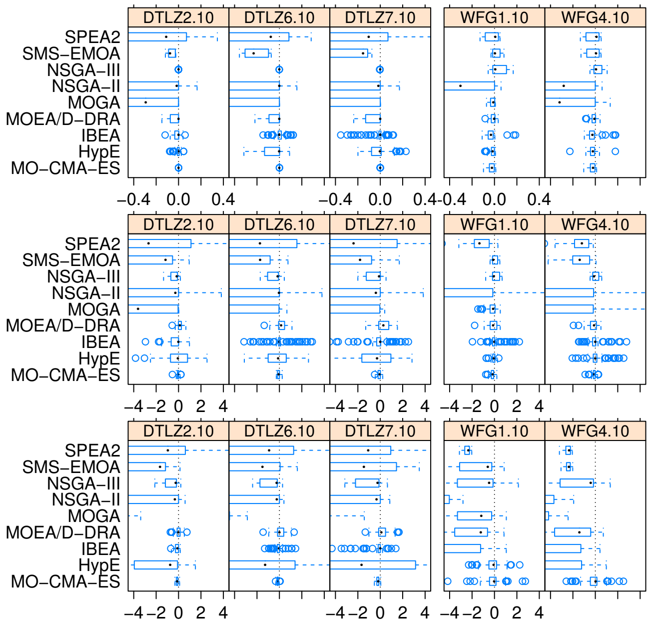

Figure 2: Performance differences between runs from MOEAs with and without unbounded archiving on the scenario with $M = 10$ and $FE_{\text{max}} = 10\,000$ for selected problems with $n_{\text{var}} = 40$. From top to bottom, I_H^{rd}, $I_{\epsilon+}$, and I_{IGD}.

results are not depicted in these plots as our goal is to contrast to the results discussed in [6], which did not consider it.

In general, runs where archiving is adopted present better results than runs without archiving. This is true for all indicators and for all number of objectives, though differences are generally more noticeable for a larger number of objectives (e.g. as shown in Figure 2 for ten objectives). Nevertheless, the extent of the benefits vary as a function of the experimental factors considered.

We next consider all FE_{max} scenarios to draw overall conclusions about performance differences. For brevity, boxplots for all metrics and scenarios are provided as supplementary material; yet, Figures 1– 2 are illustrative of the overall pattern observed. The benefits from archiving on many-objective problems when MOEAs are given $FE_{\text{max}} = 40\,000$ are rather surprising, given our runtime-constrained setup. Yet, it is likely explained by the configuration performed by irace, giving the opportunity for MOEAs to adapt to this component.

4.2 Overall state-of-the-art comparison

To assess changes caused by the adoption of unbounded archiving in the overall comparison of the state-of-the-art MOEAs, we first briefly review the most important insights from [6], which were obtained without considering the usage of archivers but using the same tuning setup:

Patterns when $M < 10$: when the number of objectives is low or moderate, MOEAs are strongly clustered into the group division of dominance-, indicator-, and decomposition-based algorithms, as discussed in Section 2, with a few exceptions. SMS and IBEA are the best performing in general, and MOGA is the worst. The performances of NSGA-II, SPEA2, and HypE worsens with the increase in M; the opposite happens for MO-CMA-ES, MOEA/D, and NSGA-III.

Disagreements between metrics: when $M = 10$, a single MOEA hardly ranks well according to all metrics—IBEA is the exception. Other than IBEA, each metric favors a different algorithm as the best among which we have SMS (I_H^{rd}), MOEA/D ($I_{\epsilon+}$), and NSGA-III (I_{IGD}).

Stopping criteria effects: since MOEAs have been configured for the different stopping criteria, their rankings do not change considerably when FE_{max} is varied. This is only false for MO-CMA-ES, which performs much better when given more FEs.

We next compare all MOEAs with unbounded archiving to discuss the changes in insights. We start with a rank sum analysis of $FE_{\text{max}} = 10\,000$ scenarios, given in Table 2. As previously discussed, I_{IGD+} results are also included. The most significant changes are as follows. The group of best-affected MOEAs comprises (i) IBEA, which is now either the best or the second best-ranked algorithm, whatever M; (ii) NSGA-II, even if it is often statistically significantly worse than the best ranked algorithm, it now ranks consistently close the group of the best-ranked ones, and; (iii) SPEA2 and HypE, which benefit the most from archiving when $M < 5$. The case of HypE is the most contrasting one, as it is among the best-ranked until M becomes large enough so that its Monte Carlo sampling component is employed ($M > 3$).[5] Conversely, the MOEA that is affected the worst in the sense of being ranked worse than before, is NSGA-III—it is now unable to rank amongst the best-ranked MOEAs, whatever M.

When all FE_{max} scenarios are considered, only some of these observations hold. To help discuss results from all 12 scenarios according to the four metrics selected, we provide a summary of the rank sum analyses in Table 3. Each cell depicts the best-ranked MOEA and the MOEAs considered statistically equivalent to it according to the given metric on the given scenario (limited to the top four ranking algorithms). The first impressive remark is the number of cells in which IBEA is present (39 out of 42). This had already been the case without archiving, but now it becomes yet more remarkable given that SMS, the other MOEA that ranked consistently well without archiving, is now unable to match IBEA results on most $FE_{\text{max}} = 40\,000$ scenarios.

The analysis provided in Table 3 further highlights the good performance of NSGA-II (on many scenarios) and of SPEA2 and HypE (scenarios with $M < 5$). Concerning NSGA-III, it once again ranks among the best-ranked according to I_{IGD}, but this time only when given enough FEs. These results from I_{IGD} are consistent with I_{IGD+} results; yet, when all scenarios are considered together, we notice subtle differences between results from these strongly correlated metrics, growing stronger as M and FE_{max} increase.

5 BOUNDED ARCHIVING

The results from unbounded archiving discussed in the previous section were surprisingly positive in the sense that when considering time-constrained scenarios as we do here, even for the largest

[5]The sampling component is the likely culprit of this behavior, as we have discussed in our previous work [6].

Table 2: Ranking of MOEAs for various metrics and values of M ($FE_{\mathbf{max}} = 10\,000$). Numbers in parenthesis give the rank-sum difference to the best ranked (left-most) MOEA. Algorithms in boldface present rank sums statistically significantly better than the rest according to Friedman's non-parametric test (99% confidence level).

					$M = 2$				
I_H^{rd}	**IBEA** (0)	**SMS** (6)	**SPEA2** (28)	NSGA-II (53)	HypE (55)	MOEA/D (104)	NSGA-III (120)	CMA (159)	MOGA (195)
$I_{\epsilon+}$	**IBEA** (0)	**SMS** (8)	**SPEA2** (28)	**NSGA-II** (34)	**HypE** (42)	MOEA/D (103)	CMA (146)	NSGA-III (151)	MOGA (181)
I_{IGD}	**SMS** (0)	**IBEA** (27)	**HypE** (32)	SPEA2 (58)	NSGA-II (100)	MOEA/D (133)	NSGA-III (185)	MOGA (204)	CMA (206)
I_{IGD+}	**SMS** (0)	**IBEA** (20)	**HypE** (40)	SPEA2 (62)	NSGA-II (100)	MOEA/D (134)	NSGA-III (182)	MOGA (206)	CMA (210)
					$M = 3$				
I_H^{rd}	**SMS** (0)	**IBEA** (10)	**HypE** (31)	NSGA-II (92)	SPEA2 (104)	MOEA/D (113)	CMA (166)	NSGA-III (180)	MOGA (230)
$I_{\epsilon+}$	**IBEA** (0)	**HypE** (8)	**SMS** (18)	NSGA-II (120)	SPEA2 (136)	CMA (171)	MOEA/D (185)	NSGA-III (205)	MOGA (209)
I_{IGD}	**HypE** (0)	**IBEA** (9)	**SMS** (37)	NSGA-II (99)	SPEA2 (142)	MOEA/D (146)	NSGA-III (197)	MOGA (210)	CMA (212)
I_{IGD+}	**IBEA** (0)	**HypE** (10)	**SMS** (24)	NSGA-II (103)	SPEA2 (149)	MOEA/D (161)	NSGA-III (196)	CMA (207)	MOGA (238)
					$M = 5$				
I_H^{rd}	**SMS** (0)	IBEA (53)	MOEA/D (98)	NSGA-II (117)	SPEA2 (143)	CMA (171)	HypE (177)	NSGA-III (188)	MOGA (258)
$I_{\epsilon+}$	**SMS** (0)	**IBEA** (10)	MOEA/D (100)	CMA (105)	NSGA-III (128)	NSGA-II (162)	SPEA2 (174)	MOGA (199)	HypE (228)
I_{IGD}	**IBEA** (0)	**SMS** (6)	NSGA-III (88)	MOEA/D (98)	CMA (99)	SPEA2 (122)	NSGA-II (143)	HypE (152)	MOGA (164)
I_{IGD+}	**SMS** (0)	**IBEA** (28)	MOEA/D (106)	NSGA-III (132)	NSGA-II (152)	SPEA2 (164)	CMA (165)	HypE (220)	MOGA (265)
					$M = 10$				
I_H^{rd}	**SMS** (0)	**IBEA** (29)	**MOEA/D** (41)	NSGA-II (69)	CMA (71)	NSGA-III (135)	HypE (149)	SPEA2 (199)	MOGA (270)
$I_{\epsilon+}$	**MOEA/D** (0)	**IBEA** (10)	NSGA-II (114)	NSGA-III (148)	SMS (162)	SPEA2 (177)	CMA (183)	MOGA (203)	HypE (218)
I_{IGD}	**IBEA** (0)	NSGA-II (88)	SPEA2 (94)	NSGA-III (142)	HypE (163)	MOEA/D (168)	CMA (170)	SMS (170)	MOGA (184)
I_{IGD+}	**IBEA** (0)	MOEA/D (66)	NSGA-II (105)	SMS (131)	NSGA-III (155)	CMA (173)	SPEA2 (193)	HypE (205)	MOGA (250)

Table 3: Summary of the statistical test results from each metric per scenario (rows: $FE_{\mathbf{max}}$; columns: M). Each cell shows MOEAs, in ranking order, that are not statistically significantly different from the best-ranked one in the cell according to Friedman's non-parametric test (99% confidence level).

	2				**3**				**5**				**10**			
	I_H^{rd}	$I_{\epsilon+}$	I_{IGD}	I_{IGD+}	I_H^{rd}	$I_{\epsilon+}$	I_{IGD}	I_{IGD+}	I_H^{rd}	$I_{\epsilon+}$	I_{IGD}	I_{IGD+}	I_H^{rd}	$I_{\epsilon+}$	I_{IGD}	I_{IGD+}
2 500	IBEA SMS	IBEA SMS	IBEA SMS	IBEA SMS	SMS IBEA	IBEA HypE SMS	IBEA HypE SMS	IBEA SMS HypE	SMS	IBEA SMS	IBEA SMS	SMS IBEA	SMS IBEA	MOEA/D	IBEA	IBEA MOEA/D
10 000	IBEA SMS SPEA2	IBEA SMS SPEA2 NSGA-II	SMS IBEA HypE	SMS IBEA HypE	SMS IBEA HypE	IBEA HypE SMS	HypE IBEA SMS	IBEA HypE SMS	SMS	SMS IBEA	IBEA SMS	SMS IBEA	SMS IBEA MOEA/D	MOEA/D IBEA	IBEA	IBEA
40 000	IBEA NSGA-II SPEA2 SMS	IBEA NSGA-II SPEA2 HypE	HypE IBEA SMS	HypE IBEA SMS	IBEA	IBEA HypE	IBEA HypE NSGA-II	IBEA	IBEA SMS	IBEA	IBEA NSGA-II	IBEA	NSGA-II MOEA/D IBEA	IBEA MOEA/D	IBEA NSGA-II NSGA-III	IBEA NSGA-III

number of objectives considered, an unbounded archive was found to be useful. In this section, we conduct additional experiments to assess scenarios when bounded archiving is used. As discussed in Section 2, we assess two different archivers, MGA and AA_s. Furthermore, we consider two different capacities, 100 and 1000 solutions, to match the population size values traditionally used for the range of objectives studied here. As a proxy to the results with bounded archivers, we truncate the results of the tuned MOEAs from Section 4 to avoid re-tuning all the MOEAs for each archiver and for each capacity of the archive; re-tuning the MOEAs for such scenarios is planned as future work.

Figures 3 and 4 show the performance differences for all MOEAs with unbounded or bounded archivers with respect to MOEAs not using archives. Results are given for the $FE_{\max} = 40\,000$ scenarios, aggregated across all problems. In more detail, Fig. 3 shows results for $M = 5$, where all archivers are tested, and Fig. 4 shows results for $M = 10$, where AA_s is not tested due to its high computational cost. In general, performance differences vary not only as a function

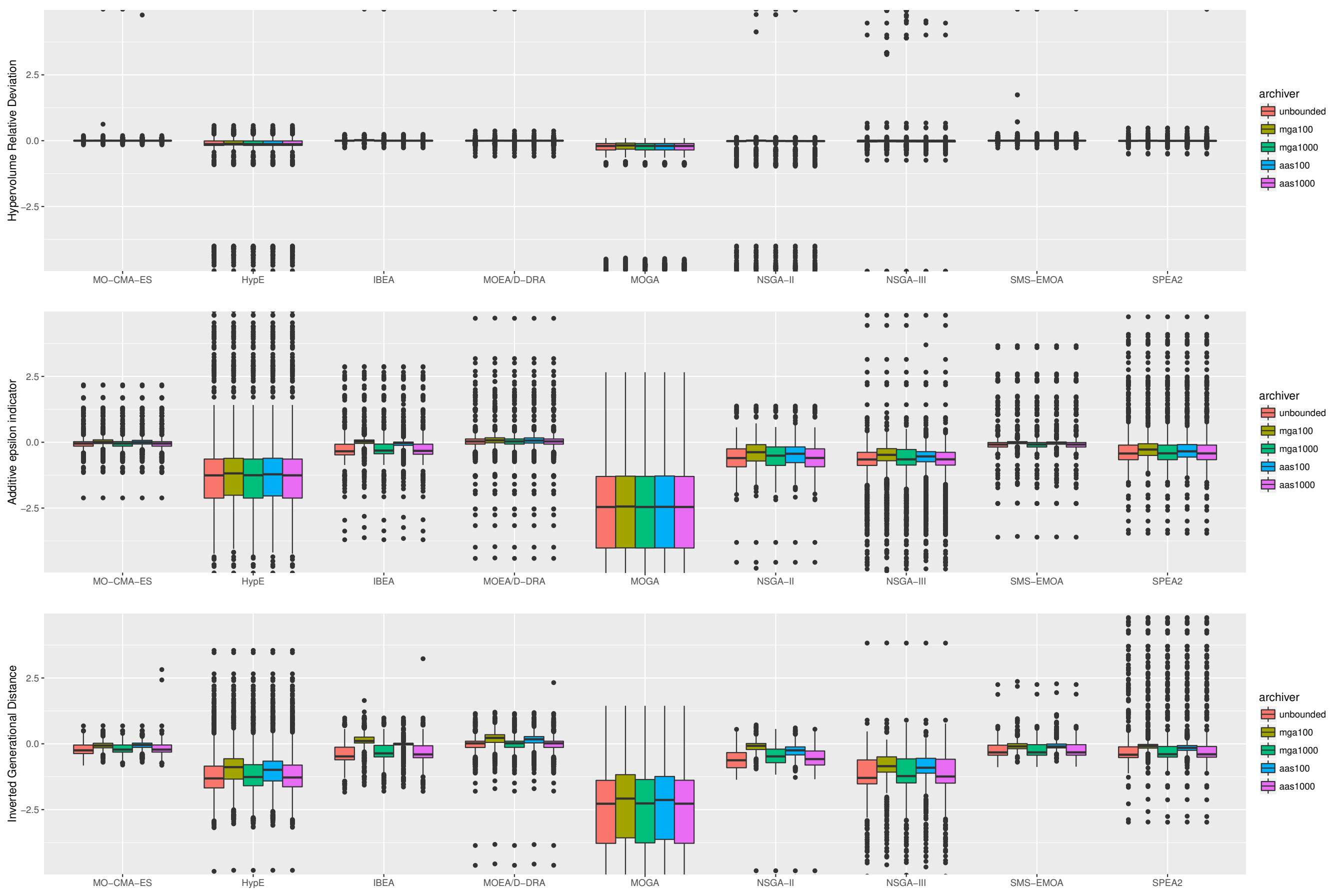

Figure 3: Boxplots of performance differences for $M = 5$ and $FE_{\max} = 40000$, grouped by MOEA and benchmark function, depicting the different archiving alternatives. From top to bottom: I_H^{rd}, $I_{\epsilon+}$, and I_{IGD}.

of M and the archiver, but also as a function of the given MOEA and performance metric. For instance, variance according to I_H^{rd} (top) plots is much smaller than for $I_{\epsilon+}$ (middle) or I_{IGD} (bottom).

Interestingly, Figure 3 indicates that the archive capacity is much more influential than the archiving method, in this case. In fact, considering the location of the boxplots of the performance differences, those for unbounded archive and an archive capacity of 1000 are typically aligned, while for an archive capacity of 100 usually less improvement is observed – independent on whether the MGA or the AA_s archiver is used. Still, an archive capacity of 100 seems to be sufficient to improve the performance of most MOEAs when compared to the tuned MOEAs without using archives, at least for five objectives. When considering the results with $M = 10$, an archive capacity of 100 leads, at least for some MOEAs and for some indicators, to worse performance than when not using archiving at all, hence indicating that an appropriate archive capacity will depend on the number of objectives.

Finally, we additionally conducted a rank sum analysis to identify which MOEAs benefit the most from archiving. Our focus in this analysis is on the unbounded archives, given that the results with a large archive capacity are close to that from an unbounded archive. No statistically significant differences are observed for any specific scenario, indicating that none of the MOEAs benefits from archiving more significantly than others. Yet, the best ranked MOEA w.r.t. improvement through archiving varies considerably from scenario to scenario. Overall, these observations indicate that a bounded archiver is a relevant (likely mandatory) component for MOEAs, specially when the number of objectives considered is large and/or the range of problems one wants to tackle is wide.

6 CONCLUSIONS

In this paper, we have analyzed how the use of archives, unbounded or bounded, affects the performance of MOEAs. Although previous works have already studied the benefits of archives [15, 22], one novelty of our analysis is that we use automatic configuration tools to tune the parameters of the considered MOEAs for the addition of an external archive, instead of assuming that there is no interaction between parameter settings and archives. Indeed, one conclusion of our study is that such interaction does exist and the presence of an archive allows MOEAs to use a much smaller population size that performs a more effective search.

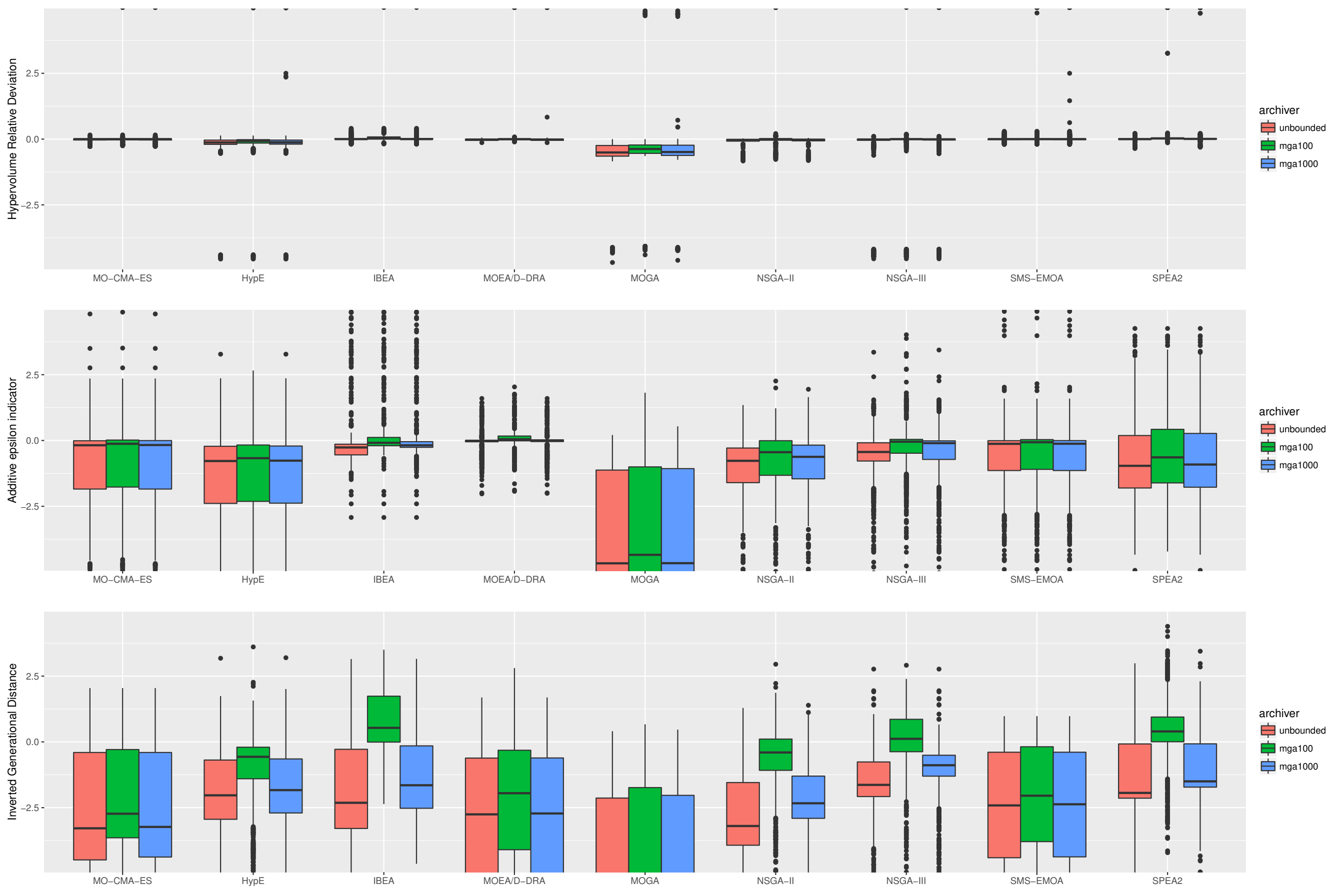

Figure 4: Boxplots of performance differences for $M = 10$ and $FE_{\max} = 40000$, grouped by MOEA and benchmark function, depicting the different archiving alternatives. From top to bottom: I_H^{rd}, $I_{\epsilon+}$, and I_{IGD}.

We have compared the effects of an unbounded archive and two variants of a bounded archive, one based on the multi-level grid archiver (MGA [23]) and the other based on the hypervolume (AA$_s$ [21]). When considering bounded archives with a rather restricted capacity, a small quality loss was observed in various scenarios, likely the reason why archives have seldom been used in MOEAs. Yet, the computational results obtained by considering unbounded archives, when compared to MOEAs without archives, generally showed improvements in quality as evaluated by four standard quality metrics. This was the case even for our time-constrained scenarios and considering the computation overhead incurred by archives, which increases with the number of objectives and the number of solution evaluations. Similar results were obtained if a bounded archive of large capacity is adopted, reinforcing the relevance of archiving already in the early stages of a MOEA design. Thus, we conclude that, unless the MOEAs must run under extremely memory-restricted conditions scenarios, it is better to use a larger external archive rather than a larger population. Moreover, the use and type of an archive must be part of the design, tuning and evaluation process, as it will influence it.

Our work can be further extended in a number of directions. The first is to consider additional types of archivers with different properties [27]. In addition, we plan to extend our experimental campaign by re-tuning all the MOEAs for bounded archives with various capacities, since our results have demonstrated the importance of configuration for MOEAs to adapt to different components. We acknowledge that our conclusions may also be rather different in the context of combinatorial optimization, where solution evaluations are often computationally very cheap and the overhead of the external archives may be more significant than in typical continuous optimization scenarios. On the other hand, there are very efficient implementations of these archiving methods, so their benefit or cost remains an open question in that context.

ACKNOWLEDGMENTS

Thomas Stützle acknowledges support from the Belgian F.R.S.-FNRS, of which he is a research director.

REFERENCES

[1] J. Bader and E. Zitzler. 2011. HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization. *Evolutionary Computation* 19, 1 (2011), 45–76.

[2] N. Beume, C. M. Fonseca, M. López-Ibáñez, L. Paquete, and J. Vahrenhold. 2009. On the complexity of computing the hypervolume indicator. *IEEE Transactions on Evolutionary Computation* 13, 5 (2009), 1075–1082.

[3] N. Beume, B. Naujoks, and M. T. M. Emmerich. 2007. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research* 181, 3 (2007), 1653–1669.

[4] L. C. T. Bezerra. 2016. *A component-wise approach to multi-objective evolutionary algorithms: from flexible frameworks to automatic design*. Ph.D. Dissertation. IRIDIA, École polytechnique, Université Libre de Bruxelles, Belgium.

[5] L. C. T. Bezerra, M. López-Ibáñez, and T. Stützle. 2017. An Empirical Assessment of the Properties of Inverted Generational Distance Indicators on Multi- and Many-objective Optimization. In *Evolutionary Multi-criterion Optimization, EMO 2017*, H. Trautmann, G. Rudolph, K. Klamroth, O. Schütze, M. M. Wiecek, Y. Jin, and C. Grimme (Eds.). Springer International Publishing, Cham, Switzerland, 31–45.

[6] L. C. T. Bezerra, M. López-Ibáñez, and T. Stützle. 2018. A Large-Scale Experimental Evaluation of High-Performing Multi- and Many-Objective Evolutionary Algorithms. *Evolutionary Computation* 26, 4 (2018), 621–656.

[7] L. C. T. Bezerra, M. López-Ibáñez, and T. Stützle. 2019. Archiver Effects on the Performance of State-of-the-art Multi- and Many-objective Evolutionary Algorithms: Supplementary material. http://iridia.ulb.ac.be/supp/IridiaSupp2019-004/. (2019).

[8] K. Bringmann and T. Friedrich. 2009. Approximating the Least Hypervolume Contributor: NP-Hard in General, But Fast in Practice. In *Evolutionary Multi-criterion Optimization, EMO 2009*, M. Ehrgott, C. M. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux (Eds.). Lecture Notes in Computer Science, Vol. 5467. Springer, Heidelberg, Germany, 6–20.

[9] C. A. Coello Coello. 2017. Recent Results and Open Problems in Evolutionary Multiobjective Optimization. In *Theory and Practice of Natural Computing - 6th International Conference, TPNC 2017*, C. Martín-Vide, R. Neruda, and M. A. Vega-Rodríguez (Eds.). Lecture Notes in Computer Science, Vol. 10687. Springer International Publishing, Cham, Switzerland, 3–21.

[10] K. Deb. 2014. Multi-objective optimization. In *Search methodologies*. Springer, 403–449.

[11] K. Deb and S. Jain. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (2014), 577–601.

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197.

[13] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. 2005. Scalable Test Problems for Evolutionary Multiobjective Optimization. In *Evolutionary Multiobjective Optimization*, A. Abraham, L. Jain, and R. Goldberg (Eds.). Springer, London, UK, 105–145.

[14] M. T. M. Emmerich and C. M. Fonseca. 2011. Computing Hypervolume Contributions in Low Dimensions: Asymptotically Optimal Algorithm and Complexity Results. In *Evolutionary Multi-criterion Optimization, EMO 2011*, R. H. C. Takahashi et al. (Eds.). Lecture Notes in Computer Science, Vol. 6576. Springer, Heidelberg, Germany, 121–135.

[15] J. E. Fieldsend, R. M. Everson, and S. Singh. 2003. Using unconstrained elite archives for multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 7, 3 (2003), 305–323.

[16] C. M. Fonseca and P. J. Fleming. 1993. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In *ICGA*, S. Forrest (Ed.). Morgan Kaufmann Publishers, 416–423.

[17] S. Huband, P. Hingston, L. Barone, and L. While. 2006. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (2006), 477–506.

[18] C. Igel, N. Hansen, and S. Roth. 2007. Covariance Matrix Adaptation for Multi-objective Optimization. *Evolutionary Computation* 15, 1 (2007), 1–28.

[19] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima. 2015. Modified Distance Calculation in Generational Distance and Inverted Generational Distance. In *Evolutionary Multi-criterion Optimization, EMO 2015 Part I*, A. Gaspar-Cunha, C. H. Antunes, and C. A. Coello Coello (Eds.). Lecture Notes in Computer Science, Vol. 9018. Springer, Heidelberg, Germany, 110–125.

[20] S. Jiang, Y. S. Ong, J. Zhang, and L. Feng. 2014. Consistencies and Contradictions of Performance Metrics in Multiobjective Optimization. *IEEE Transactions on Cybernetics* 44, 12 (2014), 2391–2404.

[21] J. D. Knowles. 2002. *Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization*. Ph.D. Dissertation. University of Reading, UK.

[22] J. D. Knowles and D. Corne. 2004. Bounded Pareto Archiving: Theory and Practice. In *Metaheuristics for Multiobjective Optimisation*, X. Gandibleux, M. Sevaux, K. Sörensen, and V. T'Kindt (Eds.). Lecture Notes in Economics and Mathematical Systems, Vol. 535. Springer, Berlin, Germany, 39–64.

[23] M. Laumanns and R. Zenklusen. 2011. Stochastic convergence of random search methods to fixed size Pareto front approximations. *European Journal of Operational Research* 213, 2 (2011), 414–421.

[24] M. López-Ibáñez, J. Dubois-Lacoste, L. Pérez Cáceres, T. Stützle, and M. Birattari. 2016. The irace package: Iterated Racing for Automatic Algorithm Configuration. *Operations Research Perspectives* 3 (2016), 43–58.

[25] M. López-Ibáñez, J. D. Knowles, and M. Laumanns. 2011. On Sequential Online Archiving of Objective Vectors. In *Evolutionary Multi-criterion Optimization, EMO 2011*, R. H. C. Takahashi et al. (Eds.). Lecture Notes in Computer Science, Vol. 6576. Springer, Heidelberg, Germany, 46–60.

[26] T. Robič and B. Filipič. 2005. DEMO: Differential Evolution for Multiobjective Optimization. In *Evolutionary Multi-criterion Optimization, EMO 2005 (Lecture Notes in Computer Science)*, C. A. Coello Coello, A. H. Aguirre, and E. Zitzler (Eds.), Vol. 3410. Springer, Heidelberg, Germany, 520–533.

[27] O. Schütze, M. Laumanns, C. A. Coello Coello, M. Dellnitz, and E.-G. Talbi. 2008. Convergence of stochastic search algorithms to finite size Pareto set approximations. *Journal of Global Optimization* 41, 4 (2008), 559–577.

[28] R. Tanabe, H. Ishibuchi, and A. Oyama. 2017. Benchmarking Multi- and Many-Objective Evolutionary Algorithms Under Two Optimization Scenarios. *IEEE Access* 5 (2017), 19597–19619.

[29] D. A. Van Veldhuizen and G. B. Lamont. 2000. Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-art. *Evolutionary Computation* 8, 2 (2000), 125–147.

[30] T. Voß, N. Hansen, and C. Igel. 2010. Improved Step Size Adaptation for the MO-CMA-ES. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2010*, M. Pelikan and J. Branke (Eds.). ACM Press, New York, NY, 487–494.

[31] L. While, L. Bradstreet, and L. Barone. 2012. A Fast Way of Calculating Exact Hypervolumes. *IEEE Transactions on Evolutionary Computation* 16, 1 (2012), 86–95.

[32] Q. Zhang and H. Li. 2007. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (2007), 712–731.

[33] Q. Zhang and P. N. Suganthan. 2009. Special Session on Performance Assessment of Multiobjective Optimization Algorithms/CEC'09 MOEA Competition. http://dces.essex.ac.uk/staff/qzhang/moeacompetition09.htm. (2009).

[34] E. Zitzler and S. Künzli. 2004. Indicator-based Selection in Multiobjective Search. In *Proceedings of PPSN-VIII, Eigth International Conference on Parallel Problem Solving from Nature*, X. Yao et al. (Eds.). Lecture Notes in Computer Science, Vol. 3242. Springer, Heidelberg, Germany, 832–842.

[35] E. Zitzler, M. Laumanns, and L. Thiele. 2002. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In *Evolutionary Methods for Design, Optimisation and Control*, K. C. Giannakoglou, D. T. Tsahalis, J. Periaux, K. D. Papaliliou, and T. Fogarty (Eds.). CIMNE, Barcelona, Spain, 95–100.

[36] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. 2003. Performance Assessment of Multiobjective Optimizers: an Analysis and Review. *IEEE Transactions on Evolutionary Computation* 7, 2 (2003), 117–132.