An empirical investigation of the optimality and monotonicity properties of multiobjective archiving methods

Li, Miqing; Yao, Xin

DOI:
10.1007/978-3-030-12598-1_2

An Empirical Investigation of the Optimality and Monotonicity Properties of Multiobjective Archiving Methods

Miqing Li[1], Xin Yao[2,1]

[1] CERCIA, School of Computer Science, University of Birmingham, Birmingham B15 2TT, United Kingdom
limitsing@gmail.com, x.yao@cs.bham.ac.uk
[2] Shenzhen Key Laboratory of Computational Intelligence (SKyLoCI), Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, P. R. China

Abstract. Most evolutionary multiobjective optimisation (EMO) algorithms explicitly or implicitly maintain an archive for an approximation of the Pareto front. A question arising is whether existing archiving methods are reliable with respect to their convergence and approximation ability. Despite theoretical results available, it remains unknown how these archivers actually perform in practice. In particular, what percentage of solutions in their final archive are Pareto optimal? How frequently do they experience deterioration during the archiving process? Deterioration means archiving a new solution which is dominated by some solution discarded previously. This paper answers the above questions through a systematic investigation of eight representative archivers on 37 test instances with two to five objectives. We have found that 1) deterioration happens to all the archivers; 2) the deterioration degree can vary dramatically on different problems; 3) some archivers clearly perform better than others; and 4) several popular archivers sometime return a population with most solutions being the non-optimal. All of these suggest the need of improvement of current archiving methods.

Keywords: multi-objective optimisation, archive, optimality, monotonicity, empirical investigation, evolutionary computation

1 Introduction

Most evolutionary multiobjective optimisation (EMO) algorithms, and other multiobjective search techniques, keep an archive[3] to capture the output of the search process. Such an archiver is typically used to approximate the Pareto front and/or as a collection of the current most promising solutions to guide next step search. Archiving can be seen as a process of taking new points from a

[3] For EMO algorithms without considering an external archive (e.g., NSGA-II [8]), their population can also be seen as an implicit archive where the selection operation is performed to preserve the best solutions ever produced [40].

point sequence, comparing them with the old points in the archive and deciding how to update the archive.

An archive of bounded size is of importance due to not only the consideration of computational resource but also search performance and later decision-making process. As such, numerous archiving methods (or archivers) emerge, known as elite preservation or environmental selection in evolutionary algorithms. They all serve the purpose of maintaining a set of well-converged and well-diversified solutions to represent the Pareto front.

However, an important issue of archiving has received relatively little attention — the optimality/monotonicity properties of archivers. In particular, one may be curious about whether an archiving method is able to return a subset of the Pareto optimal solutions discovered so far. This matters as the decision maker certainly does not want to face a situation that s/he has to select an inferior solution in the archive but misses a Pareto optimal solution once produced. In fact, many papers have observed that EMO algorithms whose archiving has no theoretical quality guarantee can suffer from dramatic performance oscillation during the search process on various instances, such as synthetic input sequences [21, 31], benchmark test problems [25, 11, 2], and real life scenarios [10, 32].

Unfortunately, most modern archivers do not have such optimality/monotonicity properties. They fail to ensure a subset of the Pareto optimal points with respect to an input sequence. Points can be preserved even when they are dominated by the points eliminated previously in the archiving process. A subsequent archive can be worse than an earlier archive. These drawbacks have been well illustrated in the literature, on different types of archiving methods, such as Pareto-based archiver [14, 25, 11], indicator-based archiver [22, 31], and decomposition-based archiver [10]. López-Ibáñez *et al.* [31] have made a comprehensive summary of the approximation properties for popular archiving methods.

On the other hand, some work focused on development of monotonic archiving methods, including theoretical analysis [14, 33, 6, 20] and algorithm design [25, 35, 18, 26]. However, without problem-specific knowledge available *a priori*, monotonic archiving methods often fail to maintain a diverse solution set and may end up with very few solutions in the archive (see [21, 31]). As such, non-monotonic archiving methods are still dominantly used everywhere.

Given the above, one interesting question raised is how current state-of-the-art archiving methods, despite their theoretical drawbacks, perform in practice. In particular, do archivers, in most cases, actually return a subset of the Pareto optimal solutions discovered so far; in other words, what percentage of solutions in the final archive are Pareto optimal? How frequently do archivers experience deterioration during the search process, in the sense that a point will still be preserved even if dominated by points which were discarded in the previous archiving? In this paper, we aim to answer the above questions. These correspond to two properties defined in [31]: 1) $\subseteq Y^*$ (i.e., the returning archive is a subset of the Pareto optimal solutions found so far) and 2) monotone (i.e., the deterioration never happens in the archiving process) We systematically investigate archiving methods associated with eight representative EMO algorithms on 37 test problems with from two to five objectives.

2 Experimental Design

2.1 Assessment Indexes

We consider two indexes, optimal ratio (OR) and deterioration ratio (DR). OR is, for one run of an EMO algorithm, the percentage of the nondominated solutions in the final archive/population are Pareto optimal with respect to all the solutions produced in this run. DR is, for one run of an EMO algorithm, the ratio of the times of deterioration occurring in the archiving process to the number of solutions considered to enter the archive, where the deterioration means that archiving a solution which is dominated by some solution discarded in the previous archiving process.

2.2 Archivers Investigated

We consider four classes of eight archiving methods: 1) Pareto-based archivers used in NSGA-II [8] and SPEA2 [41]; 2) indicator-based archivers in IBEA [39] and SMS-EMOA[4] [3]; 3) decomposition-based archivers in MOEA/D [37] and NSGA-III [7]; 4) enhanced Pareto-based archivers for many-objective optimisation (i.e., modifying Pareto dominance or density estimation) used in NSGA-II+ϵ [23] and SPEA2+SDE [28].

Pareto-based archivers first compare the Pareto dominance relation between solutions, and when the solutions have the same Pareto-based fitness (e.g., the non-dominated front in NSGA-II and the Pareto strength in SPEA2) their estimated density values are used to further distinguish between them. Indicator-based archivers adopt a performance indicator to optimise a certain preference of the solution set. In IBEA, the ϵ or dominated hypervolume indicator, based on solutions' pairwise comparison, is used, while in SMS-EMOA the set-based dominated hypervolume is used. Decomposition-based archivers decompose the space into a set of subspaces, ideally each solution representing one subspace. One difference between MOEA/D and NSGA-III is that the latter first sorts all solutions on the basis of Pareto dominance, and then decomposes the solutions on the same layer. Enhanced Pareto-based archivers increase the selection pressure of the Pareto-based archiving by either modifying the Pareto dominance criterion or modifying the crowding degree of solutions. NSGA-II+ϵ belongs to the former where the ϵ dominance [25] is used to replace crowding distance in NSGA-II, and SPEA2+SDE belongs to the latter where a position shift strategy is used to estimate solutions' density in order to make it cover both convergence and diversity.

[4] The method of computing the dominated hypervolume in SMS-EMOA was from [13], available at `http://iridia.ulb.ac.be/~manuel/hypervolume`.

2.3 Test Problems

A set of 37 problem instances were tested, including popular benchmark suites, early-developed problems and recently-developed ones. Specifically, we considered three popular suites, ZDT [38], WFG [15] and DTLZ [9]; seven early-developed problems, SCH1–SCH2 [34], FON [12], KUR [24] and VNT1–VNT3 [36]; seven recently-developed problems, convex DTLZ2 (denoted by CDTLZ2), inverted DTLZ1 (IDTLZ1), inverted DTLZ2 (IDTLZ2), scaled DTLZ1 (IDTLZ1), scaled DTLZ2 (IDTLZ2) [7, 17], multiple point distance minimisation problem (MPDMP) [23, 16], and multiple line distance minimisation problem (MLDMP) [27]. As to objective dimensionality settings of the scalable problems, the 2-objective WFG, the 3-objective DTLZ, and the 4-objective MPDMP and MLDMP (aka the rectangle problem [29]) were used; we also considered the 5-objective DTLZ1 and DTLZ2.

2.4 General Experimental Settings

All the results presented were obtained by executing 30 independent runs of each algorithm on each problem with the termination criterion of 30,000 evaluations. The population/archive size was set to 100 for all the algorithms except MOEA/D and NSGA-III where a closest number to 100 amongst the possible values was selected. To perform variation, simulated binary crossover with probability $p_c = 1.0$ and polynomial mutation with probability $p_m = 1/d$ (d denotes the number of decision variables) were considered in all the algorithms. The indicator ϵ was used in IBEA, and the PBI scalarising function was used in MOEA/D. All the parameters of the algorithms were configured as the same as in their original papers.

3 Results

3.1 Optimal Ratio

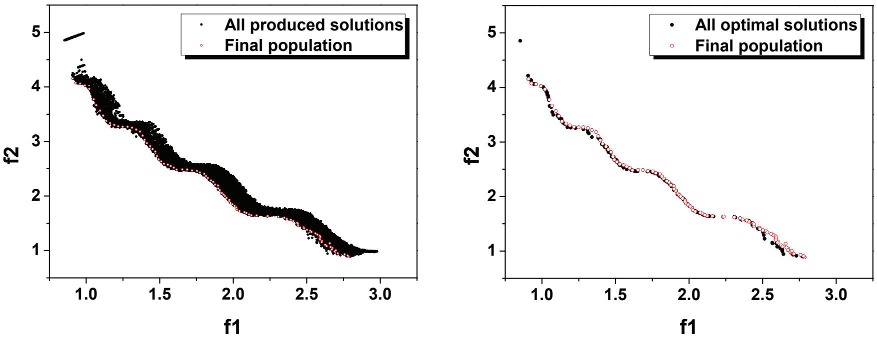
Table 1 shows the average optimal ratio (OR) of 30 runs of the eight algorithms on all the 37 problems. As can be seen, SMS-EMOA performs best, followed by SPEA2+SDE and NSGA-III; MOEA/D, NSGA-II and SPEA2 are among the worst algorithms, with only over 70% solutions being Pareto optimal[5] in their final archive on average. Taking a particular look at SMS-EMOA, unlike other algorithms whose OR varies on different problems, SMS-EMOA always achieves over 99.9% OR values on all the problems. This excellent ability may be attributed to the fact that the hypervolume value of the SMS-EMOA's archive never (or very rarely [19]) decreases and the archiving is of $\lhd$-monotonicity (see [31]) when the reference point is stable, leading to the dominated solutions hard to stay in the archive.

[5] Here, "Pareto optimal" means being nondominated to all the solutions found during the run, rather than the problem's Pareto optimal solutions.

Table 1. The average optimal ratio (OR) of 30 runs of the eight algorithms. The higher the better; 100% (in boldface) means that all the solutions in the final archive or population are Pareto optimal with respect to the produced solutions (i.e., their input sequence).

Problem	NSGA-II	NSGA-II+ϵ	SPEA2	SPEA2+SDE	IBEA	SMS-EMOA	MOEA/D	NSGA-III
SCH1	**100.0**%	99.8%	**100.0**%	99.8%	99.8%	**100.0**%	**100.0**%	**100.0**%
SCH2	**100.0**%	99.4%	99.9%	99.4%	91.0%	**100.0**%	98.4%	**100.0**%
FON	33.6%	36.5%	43.8%	60.8%	92.5%	99.4%	43.1%	86.8%
KUR	51.9%	48.1%	67.0%	69.5%	85.2%	99.9%	47.1%	75.3%
ZDT1	81.7%	86.7%	91.8%	96.4%	97.3%	**100.0**%	54.4%	99.6%
ZDT2	87.6%	91.1%	93.3%	98.4%	85.8%	**100.0**%	62.8%	99.3%
ZDT3	80.5%	86.9%	89.9%	92.1%	97.8%	99.9%	72.6%	93.5%
ZDT4	96.8%	97.2%	97.1%	99.3%	80.7%	99.9%	64.1%	96.9%
ZDT6	97.7%	98.2%	96.6%	99.6%	94.4%	**100.0**%	36.8%	98.7%
WFG1	95.7%	97.8%	98.3%	99.7%	71.8%	**100.0**%	43.2%	96.8%
WFG2	83.6%	86.8%	90.4%	92.8%	88.5%	**100.0**%	72.9%	83.9%
WFG3	61.2%	75.3%	70.3%	86.4%	93.9%	99.9%	45.0%	92.4%
WFG4	55.9%	69.1%	65.7%	83.9%	78.6%	99.7%	52.2%	91.1%
WFG5	57.8%	67.9%	72.8%	84.8%	80.1%	**100.0**%	49.0%	95.1%
WFG6	69.4%	80.6%	79.9%	92.3%	79.2%	99.9%	55.2%	93.8%
WFG7	51.3%	61.4%	60.8%	84.6%	79.0%	**100.0**%	48.9%	88.5%
WFG8	68.0%	72.0%	81.9%	94.5%	59.1%	99.8%	54.2%	85.8%
WFG9	50.8%	61.3%	55.6%	77.2%	81.0%	**100.0**%	40.4%	83.6%
VNT1	71.4%	94.7%	64.8%	96.8%	97.9%	**100.0**%	81.3%	82.3%
VNT2	56.5%	85.8%	59.0%	90.2%	91.7%	**100.0**%	67.0%	78.8%
VNT3	53.5%	37.6%	72.1%	66.0%	90.3%	99.8%	92.6%	57.5%
DTLZ1	96.2%	99.1%	90.2%	99.9%	40.7%	**100.0**%	90.1%	98.3%
DTLZ2	63.1%	69.8%	67.0%	88.9%	94.6%	**100.0**%	90.3%	77.8%
DTLZ3	94.4%	97.9%	97.6%	99.8%	27.0%	99.8%	87.3%	96.4%
DTLZ4	60.8%	71.4%	70.1%	89.8%	90.4%	**100.0**%	94.0%	78.7%
DTLZ5	58.1%	66.6%	72.0%	87.1%	80.9%	**100.0**%	97.8%	52.0%
DTLZ6	**100.0**%	99.1%	94.1%	**100.0**%	99.9%	**100.0**%	90.7%	93.9%
DTLZ7	56.2%	82.5%	67.6%	94.2%	97.1%	**100.0**%	49.8%	68.2%
CDTLZ2	59.2%	78.0%	63.8%	94.5%	97.9%	**100.0**%	84.4%	79.7%
IDTLZ1	93.3%	99.5%	97.1%	99.9%	15.9%	**100.0**%	97.7%	98.0%
IDTLZ2	64.9%	80.3%	71.6%	93.3%	97.4%	**100.0**%	96.6%	67.3%
SDTLZ1	96.2%	99.0%	93.8%	99.3%	42.9%	**100.0**%	75.1%	98.3%
SDTLZ2	60.3%	74.6%	62.5%	82.2%	94.6%	99.3%	71.6%	77.6%
MPDMP	90.0%	94.8%	85.0%	99.4%	72.8%	**100.0**%	98.9%	93.5%
MLDMP	99.5%	99.1%	98.0%	**100.0**%	74.4%	**100.0**%	82.0%	98.6%
DTLZ1-5	57.6%	99.7%	15.9%	**100.0**%	83.9%	**100.0**%	96.8%	98.3%
DTLZ2-5	66.3%	90.1%	20.2%	94.2%	98.3%	**100.0**%	91.1%	86.6%
Average	73.54%	82.05%	76.15%	91.54%	81.62%	99.93%	72.31%	87.65%

The other seven archivers do not have these desirable properties. They can reach/approach 100% OR values on some problems (e.g., SCH1, SCH2 and DTLZ6), but perform rather poorly on some other problems (e.g., FON, KUR, WFG8 and WFG9). In addition, some archivers appear to behave quite distinctly from others on a couple of problems. For example, SPEA2 and NSGA-II perform considerably worse than the other archivers on the 5-objective DTLZ1 and DTLZ2; IBEA performs on DTLZ1, DTLZ3, IDTLZ1, and SDTLZ1; MOEA/D performs on ZDT6 and WFG1. Figure 1 shows the final population obtained by MOEA/D in a typical run on WFG1 and also all the solutions produced in this run and the Pareto optimal ones. As can be seen in the figure, many solutions of the final population of MOEA/D are not Pareto optimal of the whole set of solutions produced, particularly in the bottom right and top left of the



(a) (b)

Fig. 1. Final population obtained by MOEA/D in a typical run on WFG1, coupled with (a) all the solutions produced in this run and (b) all the Pareto optimal solutions in this run.

figure, where they are dominated by some solutions which are eliminated in the archiving process of MOEA/D.

Since all the archivers (except SMS-EMOA) are the same in terms of theoretical properties, the observations of the different behaviours are from specific problems and archiving methodologies. In general, there are several situations that lead to an archive to only/mostly contain the Pareto optimal solutions. The first is that the newly produced solutions are typically dominated by some solutions in the archive. This happens on the test instance SCH1. The second situation is that a certain amount of newly produced solutions dominate some solutions in the archive even at the end of the evolution process. This happens often when the archive does not approach the Pareto front yet, such as Pareto-based algorithms on DTLZ1 and DTLZ3. The last situation is that the newly produced solutions are usually nondominated to the solutions in the archive, and also nondominated to any previously produced one. This happens either when the produced solutions are already Pareto optimal to the given problem (such as SCH2 and DTLZ6), or when they are stuck in the local optimum (such as Pareto-based algorithms on MLDMP).

Now, comparing different classes of the archiving methods, enhanced Pareto-based archivers generally outperform Pareto-based ones, with NSGA-II+ϵ and SPEA2+SDE improving the original NSGA-II and SPEA2 on average by around 10% and 15% respectively. This means that the density-based secondary archiving criterion (without incorporating convergence information) leads to OR degenerating. As to the three mainstream archiving classes, the Pareto-based, the indicator-based, and the decomposition-based, there is no clear pattern between their OR values. But we can infer the importance of Pareto dominance as the first archiving criterion in decomposition-based archiver, as NSGA-III, equipped with the Pareto nondominated sorting, performs significantly better than MOEA/D.

3.2 Deterioration Ratio

Next, let us move to the deterioration ratio (DR) results. DR denotes the ratio of the times of deterioration occurring in the archiving process to the total times of the archiving operations, and DR = 0% implies OR = 100%. It is then expected that a similar pattern to OR will be observed. Table 2 gives the average DR of 30 runs of the eight algorithms. Surprisingly, IBEA, which takes the fifth place on the average OR result, performs best here, slightly better than SMS-EMOA. One possible explanation is that the deterioration occurs mainly during the late phase of IBEA's evolutionary process, thereby some dominated solutions (in a global sense) remaining in the final archive. In contrast, in SMS-EMOA the deterioration occurs mainly during the evolutionary phase when the archive does not approach the Pareto front. This is also supported by the poor DR values of SMS-EMOA on WFG1 and DTLZ3 where the final archive is still far from the Pareto front.

It is noticed that MOEA/D reaches nearly 10% DR on average, significantly higher than the other algorithms, indicating that its archiving process preserves many dominated solutions with respect to the input sequence. This, interestingly, is contrary to the observations in [2], where the authors have seen that MOEA/D perform well (against Pareto-based, indicator-based and enhance Pareto-based EMO algorithms) in archiving the Pareto optimal solutions found on the MNK-landscape problem [1]. One possible reason for this could be different behaviours of MOEA/D between on continuous problems and on combinatorial problems. Another more likely explanation is the different performances of MOEA/D in exploration and archiving. The matting selection which considers neighbouring solutions in MOEA/D could be promising in generating Pareto optimal solutions, but it is difficult for the archive (here the population) to always keep them; i.e., good solutions can be easily generated and easily discarded as well.

3.3 Summary

Now we make a summary of the above observations.

- Consistent with the theoretical results, deterioration can happen to all the archivers in practice. However, the deterioration degree may vary dramatically on different test problems.
- SMS-EMOA performs best, especially in preserving the Pareto optimal solutions in the final archive. This is probably due to the desirable property of its hypervolume-based archiving — the $\triangleleft$-monotonicity [31]. Such a hypervolume-based bounded-size archiving, originally proposed in [22], can significantly reduce the occurrences of deterioration.
- IBEA does well in preventing the dominated solutions (with respect to the input sequence) from entering the archive, but it works mainly at the early phase of the evolution. This leads to the archive often ending up not being a subset of the Pareto optimal solutions.

Table 2. The average deterioration ratio (DR) of 30 runs of the eight algorithms. The lower the better; 0.00% (in boldface) means that there is no archived solution which is dominated by the solutions eliminated in the previous archiving process.

Problem	NSGA-II	NSGA-II+ϵ	SPEA2	SPEA2+SDE	IBEA	SMS-EMOA	MOEA/D	NSGA-III
SCH1	**0.00**%	0.10%	**0.00**%	0.08%	0.20%	**0.00**%	0.25%	**0.00**%
SCH2	**0.00**%	0.22%	**0.00**%	0.14%	0.36%	**0.00**%	0.50%	**0.00**%
FON	10.21%	10.66%	4.05%	5.13%	1.02%	0.02%	2.99%	0.66%
KUR	5.06%	6.12%	1.83%	2.05%	0.49%	0.05%	5.86%	3.74%
ZDT1	3.79%	2.78%	1.90%	0.94%	0.05%	0.67%	21.65%	0.68%
ZDT2	2.42%	1.74%	1.23%	0.47%	0.11%	0.42%	29.26%	0.77%
ZDT3	3.46%	2.42%	1.92%	1.48%	0.06%	0.85%	22.98%	2.64%
ZDT4	0.58%	0.41%	0.40%	0.21%	0.26%	0.73%	12.25%	0.53%
ZDT6	0.97%	0.67%	0.62%	0.20%	**0.00**%	0.72%	19.00%	0.60%
WFG1	1.11%	0.60%	1.13%	0.30%	**0.00**%	3.96%	16.16%	2.11%
WFG2	1.81%	1.37%	1.20%	0.75%	0.03%	1.02%	17.45%	2.64%
WFG3	7.48%	4.75%	4.22%	1.71%	0.05%	0.45%	16.89%	1.72%
WFG4	8.34%	5.25%	4.70%	2.50%	0.08%	0.45%	15.81%	3.81%
WFG5	10.46%	8.00%	5.06%	4.12%	0.09%	0.31%	18.16%	2.13%
WFG6	4.88%	3.32%	2.57%	1.49%	0.04%	0.51%	17.97%	2.26%
WFG7	10.22%	7.97%	5.86%	3.02%	0.08%	0.21%	17.29%	2.86%
WFG8	2.68%	2.36%	1.37%	0.56%	0.09%	0.56%	18.69%	2.43%
WFG9	12.04%	8.51%	6.84%	4.64%	0.48%	0.19%	13.86%	3.17%
VNT1	7.65%	2.13%	9.01%	1.38%	0.93%	**0.00**%	1.65%	6.64%
VNT2	6.44%	4.27%	4.39%	2.88%	3.07%	0.01%	2.41%	8.61%
VNT3	7.24%	10.18%	2.70%	3.94%	2.19%	0.02%	0.65%	11.85%
DTLZ1	1.25%	0.41%	1.49%	0.30%	0.55%	0.74%	2.78%	1.08%
DTLZ2	9.55%	7.45%	7.66%	2.77%	0.34%	0.08%	2.93%	3.03%
DTLZ3	1.22%	0.68%	1.10%	1.08%	0.97%	2.26%	6.44%	2.08%
DTLZ4	9.53%	7.85%	6.56%	2.52%	0.36%	0.14%	5.81%	3.57%
DTLZ5	9.11%	7.52%	5.84%	2.82%	0.17%	0.13%	5.10%	12.68%
DTLZ6	0.01%	0.50%	2.35%	0.31%	0.05%	0.66%	5.29%	2.50%
DTLZ7	10.07%	3.72%	5.77%	1.22%	0.13%	0.19%	12.14%	8.46%
CDTLZ2	9.88%	5.62%	9.34%	1.46%	0.35%	0.07%	2.63%	3.70%
IDTLZ1	1.22%	0.37%	0.68%	0.33%	0.86%	0.46%	6.13%	0.95%
IDTLZ2	9.96%	5.21%	7.11%	1.54%	0.55%	0.06%	1.38%	9.79%
SDTLZ1	1.25%	0.37%	1.38%	0.43%	0.56%	0.73%	8.81%	1.01%
SDTLZ2	9.45%	5.72%	7.62%	3.68%	0.33%	0.31%	11.58%	3.38%
MPDMP	1.17%	0.65%	1.27%	0.11%	0.01%	0.01%	0.58%	0.91%
MLDMP	0.11%	0.17%	0.28%	**0.00**%	0.08%	0.01%	0.78%	0.36%
DTLZ1-5	7.53%	0.22%	19.18%	0.05%	0.42%	0.11%	1.28%	1.47%
DTLZ2-5	12.55%	2.40%	18.23%	1.14%	0.17%	0.02%	1.13%	2.69%
Average	5.424%	3.586%	4.240%	1.561%	0.421%	0.461%	9.366%	3.176%

– Pareto-based archivers NSGA-II and SPEA2 generally perform poorly as the density-based criterion can lead to the dominated solutions frequently to enter the archive.

– Inserting convergence information into the density-based criterion of Pareto-based archivers can reduce the deteriorations. This has been shown in NSGA-II+ϵ and SPEA2+SDE.

– For indicator-based and decomposition-based archivers, Pareto dominance should still be necessary as the first criterion to select solutions. This can be inferred from the comparison between MOEA/D and NSGA-III.

4 Concluding Remarks

An archiver with theoretical quality guarantee is of high importance. It can improve search efficiency, prevent performance oscillation, and return a subset of the Pareto optimal solutions found so far. This paper has made a practical investigation of the optimality/monotonicity properties of eight representative archivers on 37 test instances. The results have shown that deterioration happens most of the time, and some archivers only return a population with less than half solutions being optimal.

It is worth pointing out that our investigation is based on the whole EMO algorithms rather than on archiving methods alone. That is, each EMO algorithm generates a different sequence of solutions that is presented to its archiving component. As such, the results (OR and DR) could be affected by the algorithm performance of producing solutions. An investigation of archiving methods under the same input sequence of solutions, independent of any EMO algorithm, can better tell their differences, which will be our next work.

Finally, note that we cannot say that a population consisting of a significantly large proportion of the current Pareto optimal solutions already well converges into the Pareto front, as it might be in the "middle" of the evolution, for example, for the Pareto-based algorithms on WFG1 and DTLZ3. But, an algorithm with a low percentage of the Pareto optimal solutions should have lots of room to be improved. In this regards, MOEA/D is an interesting example, in which good solutions can be easily generated but easily discarded as well. A combination of MOEA/D and SMS-EMOA could be potentially promising, in the sense that MOEA/D is responsible for generating solutions and updating population, while an extra archive based on the archiving method of SMS-EMOA is used to keep solutions. This would lead to different archiving methods for the different purposes in EMO — internal archiving for fostering exploration and external archiving for reducing deterioration, as suggested in [4, 5]. It is worth mentioning that a similar algorithm framework, called bi-criterion evolution [30], has been presented recently, where MOEA/D can mainly be used to generate solutions and a Pareto-based archiving method is used to keep solutions. However, this cannot prevent the occurrence of deterioration as the Pareto dominance relation and individuals' crowding degree are used to maintain the archive just like in NSGA-II and SPEA2.

Acknowledgement

This work was supported by EPSRC (Grant Nos. EP/J017515/1 and EP/P005578/1) and Science and Technology Innovation Committee Foundation of Shenzhen (Grant Nos. ZDSYS201703031748284 and JCYJ20170307105521943).

References

1. Hernán Aguirre and Kiyoshi Tanaka. Working principles, behavior, and performance of MOEAs on MNK-landscapes. *European Journal of Operational Research*, 181(3):1670–1690, 2007.
2. Hernán Aguirre, Saúl Zapotecas, Arnaud Liefooghe, Sébastien Verel, and Kiyoshi Tanaka. Approaches for many-objective optimization: Analysis and comparison on mnk-landscapes. In *International Conference on Artificial Evolution*, pages 14–28. Springer, 2016.
3. Nicola Beume, Boris Naujoks, and Michael Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007.
4. Leonardo CT Bezerra, Manuel López-Ibánez, and Thomas Stützle. Automatic component-wise design of multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 20(3):403–417, 2016.
5. Leonardo CT Bezerra, Manuel López-Ibáñez, and Thomas Stützle. A large-scale experimental evaluation of high-performing multi-and many-objective evolutionary algorithms. *Evolutionary computation*, 2018, in press.
6. David Corne and Joshua Knowles. Some multiobjective optimizers are better than others. In *The 2003 Congress on Evolutionary Computation*, volume 4, pages 2506–2512. IEEE, 2003.
7. Kalyanmoy Deb and Himanshu Jain. 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):577–601, 2014.
8. Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
9. Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable test problems for evolutionary multiobjective optimization. In Ajith Abraham, Lakhmi Jain, and Robert Goldberg, editors, *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, pages 105–145. Berlin, Germany: Springer, 2005.
10. Jonathan E Fieldsend. University staff teaching allocation: formulating and optimising a many-objective problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 1097–1104. ACM, 2017.
11. Jonathan E Fieldsend, Richard M Everson, and Sameer Singh. Using unconstrained elite archives for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 7(3):305–323, 2003.
12. Carlos Fonseca and Peter Fleming. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3(1):1–16, 1995.
13. Carlos M Fonseca, Luís Paquete, and Manuel López-Ibánez. An improved dimension-sweep algorithm for the hypervolume indicator. In *Proc. IEEE Congress Evolutionary Computation CEC 2006*, pages 1157–1163, 2006.
14. Thomas Hanne. On the convergence of multiobjective evolutionary algorithms. *European Journal of Operational Research*, 117(3):553–564, 1999.
15. Simon Huband, Philip Hingston, Luigi Barone, and Lyndon While. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5):477–506, 2006.

16. Hisao Ishibuchi, Yasuhiro Hitotsuyanagi, Noritaka Tsukamoto, and Yusuke Nojima. Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space. In *Proceedings of the International Conference on Parallel Problem Solving from Nature (PPSN)*, pages 91–100. 2010.
17. Himanshu Jain and Kalyanmoy Deb. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4):602–622, Aug 2014.
18. Huidong Jin and Man-Leung Wong. Adaptive, convergent, and diversified archiving strategy for multiobjective evolutionary algorithms. *Expert Systems with Applications*, 37(12):8462–8470, 2010.
19. Leonard Judt, Olaf Mersmann, and Boris Naujoks. Non-monotonicity of observed hypervolume in 1-greedy s-metric selection. *Journal of Multi-Criteria Decision Analysis*, 20(5-6):277–290, 2013.
20. Joshua Knowles and David Corne. Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Transactions on Evolutionary Computation*, 7(2):100–116, 2003.
21. Joshua Knowles and David Corne. Bounded pareto archiving: Theory and practice. In *Metaheuristics for multiobjective optimisation*, pages 39–64. Springer, 2004.
22. Joshua D Knowles, David W Corne, and Mark Fleischer. Bounded archiving using the lebesgue measure. In *The 2003 Congress on Evolutionary Computation*, volume 4, pages 2490–2497. IEEE, 2003.
23. Mario Köppen and Kaori Yoshida. Substitute distance assignments in NSGA-II for handling many-objective optimization problems. In *Evolutionary Multi-Criterion Optimization*, pages 727–741. Springer Berlin / Heidelberg, 2007.
24. Frank Kursawe. A variant of evolution strategies for vector optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 193–197. Springer, 1990.
25. Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*, 10(3):263–282, 2002.
26. Marco Laumanns and Rico Zenklusen. Stochastic convergence of random search methods to fixed size Pareto front approximations. *European Journal of Operational Research*, 213(2):414–421, 2011.
27. Miqing Li, Crina Grosan, Shengxiang Yang, Xiaohui Liu, and Xin Yao. Multiline distance minimization: A visualized many-objective test problem suite. *IEEE Transactions on Evolutionary Computation*, 22(1):61–78, 2018.
28. Miqing Li, Shengxiang Yang, and Xiaohui Liu. Shift-based density estimation for Pareto-based algorithms in many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 18(3):348–365, 2014.
29. Miqing Li, Shengxiang Yang, and Xiaohui Liu. A test problem for visual investigation of high-dimensional multi-objective search. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, pages 2140–2147, 2014.
30. Miqing Li, Shengxiang Yang, and Xiaohui Liu. Pareto or non-pareto: Bi-criterion evolution in multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):645–665, 2016.
31. Manuel López-Ibáñez, Joshua Knowles, and Marco Laumanns. On sequential online archiving of objective vectors. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 46–60. Springer, 2011.

32. Patrick M Reed, David Hadka, Jonathan D Herman, Joseph R Kasprzyk, and Joshua B Kollat. Evolutionary multiobjective optimization in water resources: The past, present, and future. *Advances in water resources*, 51:438–456, 2013.
33. Günter Rudolph and Alexandru Agapie. Convergence properties of some multi-objective evolutionary algorithms. In *Proceedings of the 2000 Congress on Evolutionary Computation*, volume 2, pages 1010–1016. IEEE, 2000.
34. J David Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, pages 93–100, 1985.
35. Oliver Schütze, Marco Laumanns, Carlos A Coello Coello, Michael Dellnitz, and El-Ghazali Talbi. Convergence of stochastic search algorithms to finite size pareto set approximations. *Journal of Global Optimization*, 41(4):559–577, 2008.
36. R Vlennet, Christian Fonteix, and Ivan Marc. Multicriteria optimization using a genetic algorithm for determining a pareto set. *International Journal of Systems Science*, 27(2):255–260, 1996.
37. Qingfu Zhang and Hui Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.
38. Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195, June 2000.
39. Eckart Zitzler and Simon Künzli. Indicator-Based Selection in Multiobjective Search. In *Proceedings of the International Conference on Parallel Problem Solving from Nature (PPSN)*, pages 832–842. 2004.
40. Eckart Zitzler, Marco Laumanns, and Stefan Bleuler. A tutorial on evolutionary multiobjective optimization. In *Metaheuristics for multiobjective optimisation*, pages 3–37. Springer, 2004.
41. Eckart Zitzler, Marco Laumanns, and Lothar Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design, Optimisation and Control*, pages 95–100, Barcelona, Spain, 2002.