Is our archiving reliable? Multiobjective archiving methods on “simple” artificial input sequences

Li, Miqing

DOI:
10.1145/3465335

Is Our Archiving Reliable? Multiobjective Archiving Methods on "Simple" Artificial Input Sequences

MIQING LI, University of Birmingham, U. K.

In evolutionary multi-objective optimisation (EMO), archiving is a common component that maintains an (external or internal) set during the search process, typically with a fixed size, in order to provide a good representation of high-quality solutions produced. Such an archive set can be used solely to store the final results shown to the decision maker, but in many cases may participate in the process of producing solutions (e.g., as a solution pool where the parental solutions are selected). Over the last three decades, archiving stands as an important issue in EMO, leading to the emergence of various methods such as those based on Pareto, indicator or decomposition criteria. Such methods have demonstrated their effectiveness in literature and have been believed to be good options to many problems, particularly those having a regular Pareto front shape, e.g., a simplex shape.

In this paper, we challenge this belief. We do this through artificially constructing several sequences with extremely simple shapes, i.e. 1D/2D simplex Pareto front. We show the struggle of predominantly-used archiving methods which have been deemed to well handle such shapes. This reveals that the order of solutions entering the archive matters, and that current EMO algorithms may not be fully capable of maintaining a representative population on problems with linear Pareto fronts even in the case that all of their optimal solutions can be found.

CCS Concepts: • **Mathematics of computing → Combinatorics**; **Combinatoric problems**; • **Applied computing → Multi-criterion optimization and decision-making**.

Additional Key Words and Phrases: multi-objective optimisation, archiving, elitism, population maintenance, deterioration

1 INTRODUCTION

Two pivotal issues centred around evolutionary multiobjective optimisation (EMO) are solution generation and population maintenance. The first one is concerned with mating selection and variation (e.g., crossover and mutation) where we want to produce promising offspring, hopefully better than their parents. The other is concerned with environmental selection (aka elitism) where we want to avoid losing the very best solutions found during the search course. The second issue can also be generalised as archiving, a process of taking new solutions, comparing them with the old ones and deciding how to update the population/archive [47].

Archiving in EMO can be implemented in different ways. It can be done implicitly by combining the old population and its offspring and then reducing them to the next population (such as in NSGA-II [13]), or by maintaining an internal archive which helps produce high-quality solutions (such as in SPEA2 [63]), or by explicitly maintaining an external archive for storing high-quality solutions produced so far (such as in MOEA/D [60]). On the other hand, there are various theoretical/empirical studies exclusively on archiving in the field, such as [4, 7, 30, 32, 38, 43, 47, 48, 53, 54]. They have shown the importance of the archiving operation and its challenge faced in practice.

Since memory resources are usually restricted, criteria/rules have to be set to decide which solutions are kept and which are removed. In general, there exist three classes of representative criteria in archiving: Pareto-based criteria, indicator-based criteria and decomposition-based criteria. Pareto-based criteria consider the Pareto dominance relation between solutions, and when solutions are incomparable with respect to Pareto dominance, density information is introduced to distinguish between them. Indicator-based criteria define a quality indicator, and compare solutions through

Author's address: Miqing Li, m.li.8@bham.ac.uk, University of Birmingham, Birmingham, U. K., B15 2TT.

their contributions to the indicator. Decomposition-based criteria often associate solutions with reference vectors (points/rays), and compare their quality on the basis of specific reference vectors. It is well known and studied in the field that the performance of archiving with these criteria is significantly affected by properties of a given problem's Pareto front (e.g., see [27, 40]). For example, archiving with Pareto-based criteria typically performs poorly on a Pareto front with high dimensions [51]; archiving with indicator-based criteria may favour knee areas of a Pareto front [17]; and archiving with decomposition-based criteria usually struggle on an irregular Pareto front shape [45].

Nevertheless, it is commonly believed that all the three classes of criteria work well on low-dimensional Pareto fronts with simplex shapes, e.g., a linear/triangular shape [27, 42]. In this paper, we challenge this belief. We construct several sequences[1] of 2D/3D points (solutions) with the simplest Pareto front shapes, but with "interesting" orders of the points fed to the archiver. For example, the set of the points fed gradually expands/shrinks, or there are some early-coming Dominance Resistant Solutions (DRS)[2] in the sequence. We construct seven such sequences, and show some interesting but disappointing observations from five well-established archiving methods (i.e., those from NSGA-II [13], IBEA [62], SMS-EMOA [2], MOEA/D [60] and NSGA-III [11]) as well as one more advanced archiving method MGA [38].

It is worth mentioning that the nature of optimisation problems (along with randomness in evolutionary search) may lead to various sequences of solutions. For example, some test problems (e.g., KUR [36], the UF suite [61] and the problems constructed in [46]) typically lead to the search of an EMO algorithm in the objective space to start from a particular region, and some other test problems (e.g., DTLZ3 [14] and ML-DMP [40]) often result in the generation of DRS solutions during the search. In addition, there exist many algorithms which tend to find solutions in a certain order[3]. For example, the algorithm in [50] starts the search from one extreme solution (i.e., the best solution in one of the two objectives) and then gradually moves to the other. The algorithm in [15] searches for solutions back and forth between the two objectives. The algorithms in [49] and [22] were designed to find all the extreme solutions of a problem's Pareto front first and then inner solutions. Note that such practices are particularly common in conventional mathematical programming since they typically compute solutions one by one [10, 55]. For example, to solve the bi-objective minimum spanning tree problem, the algorithm in [55] first computes the two extreme solutions and then a particular inner solution; these two steps are repeated by treating the inner solution as a new extreme one until all Pareto optimal solutions are found.

Given the above, an independent investigation of archiving methods in handling various solution sequences is worthwhile. Being immune from the interference of randomness in the search, it tells us how reliable multiobjective optimisation algorithms are in maintaining high-quality solutions found. In addition, we would like to clarify that the sequences presented here are not designed to simulate various real-world scenarios, but to challenge well-established EMO archiving methods on even simple Pareto fronts.

[1] A sequence of points is an enumerated collection of points in which order matters. Archiving for a sequence of points is different from for a set of points as in the latter the selection operation is conducted after all the points arrived. Archiving for a point sequence is an online process in which the archiver does not know what point is coming next.

[2] Dominance resistant solutions are those with a very poor value in one objective but with (near) optimal values in the others [23].

[3] For multi-objective problems whose single-objective version can be solved polynomially like TSP, the search may not be executed in a certain order (since one can directly generate supported solutions and then fill the gaps between them, namely, working on non-supported solutions).

2 PRELIMINARY

Multi-objective optimisation problems (MOPs) are mathematical optimisation problems involving more than one objective to be minimised/maximised. Without loss of generality, in this paper we consider a minimisation MOP with m objective functions $f : X \to Z \subset \mathbb{R}^m$. The objective functions map a decision vector $\mathbf{x} \in X$ in the decision space to a vector $f(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_m(\mathbf{x}))$ in the objective space. Our study focuses on archiving objective vectors. For simplicity, we refer to an objective vector as a *solution* (or a *point*).

Considering two solutions $\mathbf{z}^1, \mathbf{z}^2 \in Z$, solution $\mathbf{z}^1$ is said to *(Pareto) dominate* $\mathbf{z}^2$ (denoted as $\mathbf{z}^1 \prec \mathbf{z}^2$) if $\mathbf{z}^1_i \leq \mathbf{z}^2_i$ for $1 \leq i \leq m$ and there exists at least one objective j on which $\mathbf{z}^1_j < \mathbf{z}^2_j$. A solution $\mathbf{z} \in Z$ is called *Pareto optimal* if there is no solution in Z that dominates $\mathbf{z}$. The set of all Pareto optimal solutions in Z of an MOP is called its *Pareto front*.

As the size of an MOP's Pareto front is usually prohibitively large (or even infinite), one may be interested in finding an approximation set of it with a (fixed) manageable size. This is particularly useful for population-based search (e.g., evolutionary computation), where the search population, typically consisting of a fixed number of members, is used to approximate the Pareto front gradually. As such, rules/criteria have to be set to decide how to update the approximation set on the fly, i.e., which solutions are kept and which are removed with the introduction of new solutions. This update process can be generalised as *archiving*, a process to deal with a sequence of solutions with no knowledge of future inputs [31, 47].

Note that although we here consider a sequence of discrete Pareto optimal points, the Pareto front of an MOP can be continuous. For such problems, the size of the Pareto front is necessarily infinite. A tolerance level should be used to make it practically manageable. However, setting a proper tolerance level may not be easy; it depends on the properties of the problem in hand (e.g., its objective dimensionality and the correlation between the objectives) [5].

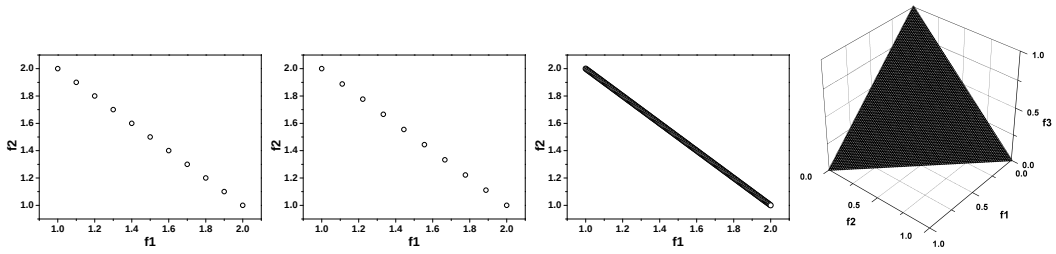
3 TEST SEQUENCES

To the best of our knowledge, there exist two studies in the literature [30, 47] which artificially constructed point sequences to test archiving methods. They designed several sequences, with an intention of showing that commonly-used archiving methods suffer from keeping a point that is dominated by some point eliminated previously in the archiving process, a problem called deterioration [37].

In this study, apart from the deterioration, we also investigate the effect of sequences with different patterns of the points on the Pareto front (e.g. expanding and shrinking). More importantly, we consider sequences with the simplest Pareto front shapes, which it is long believed that current archiving methods are capable of handling. The Pareto optimal solutions of the considered seven sequences[4] are given in Figure 1. As can be seen, Sequences 1 and 2 only have a handful of Pareto optimal solutions, whilst the rest have hundreds and thousands. The last four sequences share the same 3D Pareto front. Minimisation of objectives is considered for all the sequences.

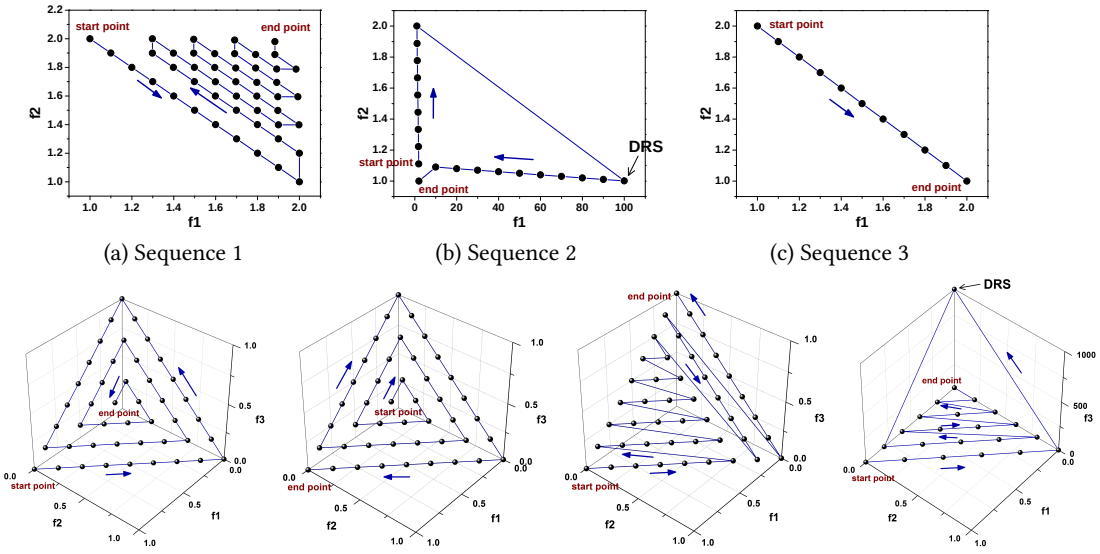
The main challenges of the constructed sequences lie in the order of points fed to the archives, which is shown in Figure 2. Sequence 1 is comprised of points with generally declining quality, (i.e., the quality of early-coming points is generally better than that of late-coming ones), but the capacity of the archives is set not big enough to store all Pareto optimal points. Sequences 2 and 7 have one DRS point which appears early in the sequence but only is dominated by very last points of the sequence. The points in Sequence 3 shift from one corner to the other corner. The points in Sequence 4 shrink from outside to inside, the points in Sequence 5 do exactly the opposite. The points in Sequence 6 go along a zigzag line, and with a turn in the middle. The Sequences 3–6

[4]The sequences considered in our study are available at https://www.cs.bham.ac.uk/~limx/Data/Sequences.rar.



(a) Sequence 1 (b) Sequence 2 (c) Sequence 3 (d) Sequences 4–7

Fig. 1. Pareto optimal solutions of the seven point sequences.



(a) Sequence 1 (b) Sequence 2 (c) Sequence 3

(d) Sequence 4 (e) Sequence 5 (f) Sequence 6 (g) Sequence 7

Fig. 2. Illustrations of the seven sequences of points, where Sequences 1 and 2 are actual ones considered and Sequences 3–7 are downsized ones (i.e., with fewer points) for clearer viewing. The blue arrow indicates the order of the points fed to the archiver. For Sequences 2 and 7, the Dominance Resistant Solution (DRS) is labelled. For Sequence 1, points arrives with generally declining quality. For Sequence 2, please note the difference of the scale on the objective f_1 between Fig. 2(b) and Fig. 1(b), and the leftmost ten points in Sequence 2 in Fig. 2(b) are nondominated to each other which are the Pareto optimal solutions shown in Fig. 1(b).

challenge archivers in maintaining a set of diverse solutions when the search moves on from one region to another along the Pareto front.

4 ARCHIVERS INVESTIGATED

We first consider three classes of five well-established archiving methods: 1) a Pareto-based archiver which was used in NSGA-II [13]; 2) indicator-based archivers in IBEA [62] and SMS-EMOA [2]; 3) decomposition-based archivers in MOEA/D [60] and NSGA-III [11]. Afterwards, we consider a

Table 1. Settings of the sequences and archivers.

Sequence	Dimension	No. of points	No. of Pareto optimal points	Archive capacity	h in MOEA/D, NSGA-III
1	2D	47	11	10	9
2	2D	20	10	10	9
3	2D	201	201	10	9
4	3D	5151	5151	105	13
5	3D	5151	5151	105	13
6	3D	5151	5151	105	13
7	3D	20302	20301	105	13

state-of-the-art archiver, multi-level grid archiver (MGA) [38], to see how well a very advanced archiver performs on these sequences.

The Pareto-based archiver in NSGA-II sorts solutions into layers according to the Pareto dominance relation. When the archive cannot accommodate all nondominated solutions in a particular layer, a density metric, called crowding distance, is used to distinguish between them in order to keep well-distributed solutions. The indicator-based archivers IBEA and SMS-EMOA use an indicator to measure the quality of a solution set; specifically, IBEA considers either the ϵ [37] or hypervolume [64] indicator while SMS-EMOA considers the hypervolume indicator. A major difference between IBEA's archiver and SMS-EMOA's is that the former considers pairwise comparison whilst the latter considers set-based comparison. The decomposition-based archivers in MOEA/D and NSGA-III decompose the objective space into a set of subspaces, ideally each solution corresponding to one subspace. A notable difference of NSGA-III's archiver to MOEA/D's is that NSGA-III inherits the non-dominated sorting strategy of NSGA-II; that is, first sort all solutions on the basis of Pareto dominance and then decompose the solutions on the same layer. The same applies to SMS-EMOA. Note that whether or not Pareto dominance sorting is used first in the archiving process will not affect the results here since the new point fed to the archive is set to be always nondominated to the current archive. MGA compares solutions by using a hierarchical grid to define a family of ϵ-dominance relations. To accept a new solution, it uses a function that counts the number of occupied boxes (on various levels). It can maintain a monotonous progress to a solution set that covers the Pareto front with non-overlapping boxes at finest resolution possible [38].

It is necessary to note that some of the algorithms investigated (e.g. NSGA-II and IBEA) use the batch-wise population/archive update mechanism (i.e., $\mu + \mu$). That is, they only update the population/archive after there are μ new solutions produced/arriving. Here, we adopted the one-by-one update mechanism (i.e., $\mu + 1$) following the practice in [43, 47]. That is, to update the archive once a new point arrives. As such, the conclusions drawn from the experiments apply only to point-by-point archivers. In addition, for convenience we may sometimes use the name of an EMO algorithm to represent its archiver in result descriptions.

Some of the archivers need configurations, e.g., IBEA and MOEA/D. Here, all the parameters were configured as the same as in their original papers unless expressly stated otherwise. In MOEA/D, the Tchebycheff scalarising function was used. The reference point in MOEA/D was checked for update by the new point before performing the archiving. That is, if the value of any objective of the new point is better (i.e., smaller) than that of the reference point, then update the corresponding objective of the reference point by that value. During the archiving process, all the current points in the archive were compared with the new point for the replacement (i.e. the whole archive is considered as the neighbourhood in MOEA/D). As for IBEA, the quality indicator ϵ was used and the scaling factor κ was 0.05.

In order to challenge the archivers, we intentionally set the capacities of the archives to those that are not very "comfortable" for the archivers to handle. For example, in Sequence 1, the number of the Pareto optimal points is 11, and we then set the archive capacity to be 10, so that archivers cannot determine the points solely by Pareto dominance. Yet, we do consider specific requirements of some archivers (i.e. the decomposition-based archivers of MOEA/D and NSGA-III); we set the archive capacity so that a set of perfectly distributed points can be maintained if done properly. Overall, our intention is to make the performance of archivers least affected by other factors but the order of the points in the sequences. Table 1 summarises the settings of the sequences and archivers.

5 RESULTS

In this section, we first report the results of the five widely-used archiving methods, i.e., those in NSGA-II, IBEA, SMS-EMOA, MOEA/D and NSGA-III. This is followed by the results of the state-of-the-art archiver MGA. Lastly, we summarise those results.

Figures 3, 5, 6, 7, 9, 11 and 13 show the final solution sets obtained by the five archivers on the seven sequences. For quantitative comparison, we also give the evaluation results of the solution sets on the quality indicator *hypervolume* [64]. The hypervolume indicator of a solution set calculates the volume of the union of the hypercubes determined by each of its solutions and a reference point. It is a comprehensive indicator which can (partially) reflect four quality aspects of a solution set, convergence, spread, uniformity and cardinality [44]. Note that here the Pareto fronts of the sequences considered are of simplex shapes; there is no knee point, and hypervolume will not prefer particular points on the Pareto front. In addition, due to the simplex Pareto front, according to [25], it is appropriate to set the reference point as $nadir_i + l_i/h$, where $nadir_i$ is the nadir point of the Pareto front on its ith objective, l_i is the range of the Pareto front on the ith objective, and h is an integer subject to $C_{m-1}^{h+m-1} \leq n < C_{m-1}^{h+m}$ (m and n being the number of objectives and the size of the solution set, respectively). That is, the reference point $(2 + 1/9, 2 + 1/9)$ for the bi-objective sequences, and $(1 + 1/13, 1 + 1/13, 1 + 1/13)$ for the tri-objective ones. Here, we report the ratio of the hypervolume value of the obtained solution set to the Pareto front (i.e., a collection of all the nondominated solutions of the sequence), denoted by HVR. Table 2 gives the HVR results of the five archivers on the seven sequences.

It is worth mentioning that there exist other quality indicators that can also reflect comprehensive quality of a solution set, such as IGD [9]. Evaluation results obtained by them may not be exactly the same as those obtained by HVR, since quality indicators may not "agree" with each other, particularly between Pareto compliant indicators and non-Pareto compliant ones. Fortunately, here all the fronts of the sequences are of simplex shapes, in which case the results obtained by different indicators tend to much more consistent [3, 29].

Table 3 summarises the quality of the solution sets obtained by the archivers on the seven sequences, where their quality has been classified into five levels, excellent, good, fair, poor, and very bad. As it can be seen, there is no archiver that is able to achieve a reasonable level (good or excellent) on all the sequences, or even on a majority of the sequences. For some sequences, it is even hard for many archivers to achieve the fair level, such as Sequences 2, 6 and 7. In the following subsections, we will detail how the archivers behave on each sequence.

5.1 Sequence 1 (Figure 3)

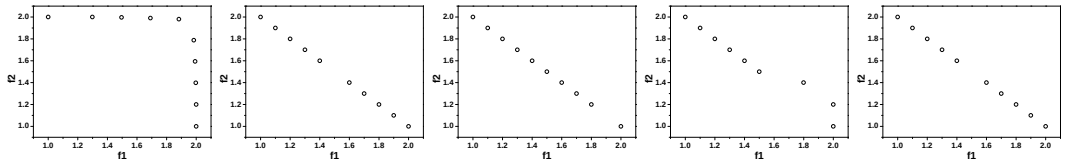
As shown in Figure 2(a), the points of Sequence 1 come with declining quality. The sequence can easily be maintained well by most of the investigated archivers as shown in Figure 3, but certainly not by NSGA-II. The point set of NSGA-II is the worst possible nondominated set obtained from this sequence.

Table 2. The HVR value of the five widely used archivers on the seven sequences.

Sequence	NSGA-II	IBEA	SMS-EMOA	MOEA/D	NSGA-III
1	35.714%	98.539%	98.540%	88.335%	98.539%
2	94.563%	34.633%	76.409%	89.089%	83.658%
3	92.207%	89.349%	89.506%	77.017%	86.480%
4	92.311%	83.717%	95.092%	96.517%	96.681%
5	95.120%	96.251%	96.267%	94.797%	94.600%
6	92.880%	95.338%	95.012%	93.647%	96.011%
7	89.649%	22.569%	60.165%	95.958%	73.318%

Table 3. Properties of the test sequences and the behaviour of the five archivers on them.

Sequence	Dimensionality	Characteristic	NSGA-II	IBEA	SMS-EMOA	MOEA/D	NSGA-III
1	2D	deterioration	very bad	excellent	excellent	fair	excellent
2	2D	DRS	fair	very bad	poor	good	poor
3	2D	shifting	good	fair	fair	poor	poor
4	3D	shrinking	fair	very bad	fair	excellent	excellent
5	3D	expanding	fair	good	good	poor	poor
6	3D	shifting	poor	poor	poor	poor	fair
7	3D	DRS	poor	very bad	very bad	good	very bad

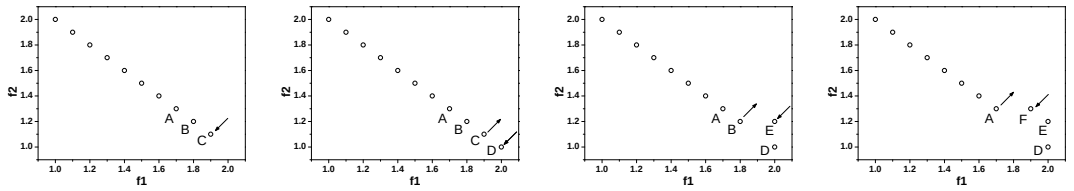


(a) NSGA-II HVR=35.714% (b) IBEA HVR=98.539% (c) SMS-EMOA HVR=98.540% (d) MOEA/D HVR=88.335% (e) NSGA-III HVR=98.539%

Fig. 3. The final point sets obtained by the five archiving methods on Sequence 1 and their corresponding HVR results.

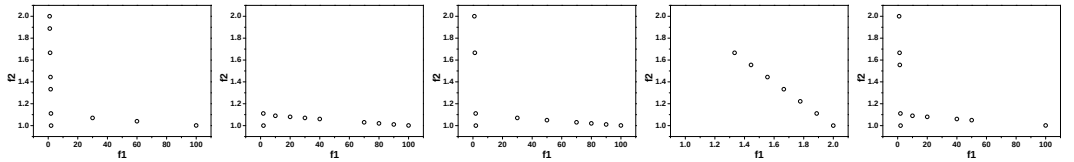
To understand why this happened, we give several consecutive iterations of the step-by-step archiving process of NSGA-II in Figure 4. First, the point **C** is fed to and accepted by the archive in Figure 4(a), and then the archive reaches its maximum capacity (10). Next, **D** is fed to the archive (Figure 4(b)), and then one of the 11 nondominated points will be eliminated. Note that we set the distance between points **C** and **D** to be slightly smaller than that between the other adjacent points. So, according to the crowding distance [13] the point **C** will be eliminated since **D** is the extreme point. Next, **E** is fed to the archive (Figure 4(c)). As **C** has been already eliminated, there is no point in the current archive which dominates **E**. Note that points **B** and **D** do not dominate **E** — we intentionally set that they are very close to but actually not dominate **E** (i.e., **E** is slightly better than **B** on the objective f_2 and than **D** on the objective f_1). Now, all the 11 points are mutually nondominated, and one of them will be eliminated. Point **B** is that point since it has the smallest crowding distance (though the margin is tiny). In Figure 4(d), likewise, point **F** enters the archive and edges out point **A**. This archiving process continues and the procedure ends with the final archive containing very poor quality points, as shown in Figure 3(a).

In fact, such a deterioration phenomenon has been frequently observed and studied since early 2000s, e.g., in [19, 32, 37, 52]. The reason for this occurrence is that the archiver cannot prevent



(a) **C** fed to the archive (b) **D** fed to the archive (c) **E** fed to the archive (d) **F** fed to the archive

Fig. 4. Several consecutive iterations of the archiving process of NSGA-II. Note that points **B** and **D** are very close but not to dominate **E** (i.e., **E** is slightly better than **B** on the objective f_2 and than **D** on the objective f_1).



(a) NSGA-II HVR=94.563% (b) IBEA HVR=34.633% (c) SMS-EMOA HVR=76.409% (d) MOEA/D HVR=89.089% (e) NSGA-III HVR=83.658%

Fig. 5. The final point sets obtained by the five archiving methods on Sequence 2 and their corresponding HVR results.

the entry of new points that are dominated by some points eliminated previously. In the extreme situation, this may lead to the whole archive deteriorating gradually with time, as shown in our example. In addition, it is necessary to mention that the deterioration also applies to other archiving methods, e.g., the indicator-based [1, 33, 47] and decomposition-based [8, 18]. Readers who are interested in this respect can refer to related theoretical [32, 47] and empirical [43] work.

Consider the HVR values of the five archivers. NSGA-II and MOEA/D are expected to receive lower values as their Pareto optimal points are fewer than those of the other three archivers. Only two points preserved by NGSA-II are on the Pareto front of the sequence, and only seven points preserved by MOEA/D, compared to the fact that all the 10 points preserved by the other three archivers are on the Pareto front. Amongst the other three point sets, IBEA and NSGA-III have the same HVR value (as their solution sets are the same), and SMS-EMOA has a marginally higher value. The reason is as follows. The setting of the reference point and the simplex shape of the sequence's Pareto front make all Pareto optimal points (when they are perfectly uniformly distributed) have the same contributions to the hypervolume result [25]. Here, the point gap (bottom-right) of SMS-EMOA is marginally shorter than that (middle) of IBEA and NSGA-III. As a result, the HVR value received by SMS-EMOA is marginally higher than that by IBEA and NSGA-III.

5.2 Sequence 2 (Figure 5)

Sequence 2 challenges archiving methods via introducing a DRS point in the middle of the archiving process which is the best on the objective f_2 so far but extremely poor on the objective f_1 compared with other points (see Figure 2(b)). After the introduction of the DRS point, some points that are nondominated to it arrive sequentially (i.e., with better f_1 value and worse f_2 value), until the last point which dominates all the points since the DRS is introduced.

This sequence poses a big challenge to all the archivers. As shown in Figure 5, none of them is able to preserve all the 10 Pareto optimal points (cf. Figure 1(b)). For MOEA/D, there exist some

duplicate points, i.e., multiple weights corresponding to one point. The other four archivers share the same pattern, with the DRS point and some dominated points always in the archive. The reason for this is that effectively all the points in the sequence except the last one are nondominated with each other. Those early-coming "true" Pareto optimal points may be favoured less than the DRS point as well as the following points by the archivers (as their crowding degree is estimated higher than the late comers'), thus being edged out from the archive. This becomes prominent in IBEA as the algorithm favours points with bigger differences (but without considering the normalisation among objectives).

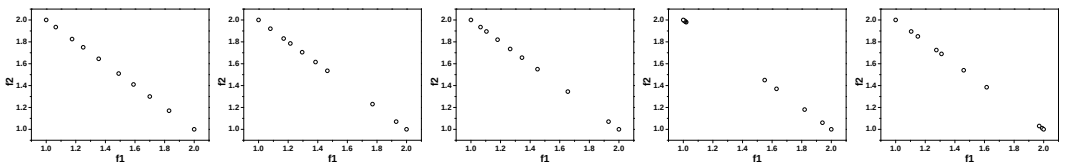
5.3 Sequence 3 (Figure 6)

As shown in Figure 2(c), the points in Sequence 3 arrive from one extreme to another. We can see from Figure 6 that the sequence has little effect on the Pareto-based archiver in NSGA-II, but more on the indicator-based and decomposition-based archivers. This occurrence is due to the fact that the density estimator in the Pareto-based archiver, which is entirely based on the distance between points, is affected very little, if at all, by the order of the points fed to the archive. In contrast, the other archivers need some reference in the calculation of their criteria, which may be affected significantly by the order of the points fed. For example, in the process of archiving Sequence 3, the reference point used to calculate the hypervolume indicator in SMS-EMOA and the ideal point used to calculate the Tchebycheff scalarising function in MOEA/D keep changing with any new point fed to the archive (since it enlarges the range of the points fed so far). This dynamic may lead to a big difference between the quality of the archive instantly and its quality at the end of the archiving process.

5.4 Sequence 4 (Figure 7)

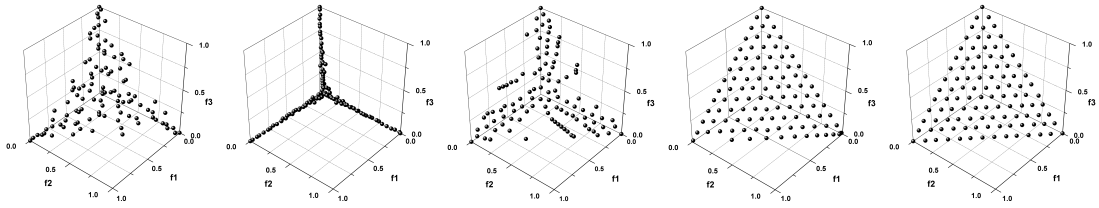
The shape of Sequence 4 is a two-dimensional simplex, and the order of the points fed into the archive is from outer triangle layers to inner triangle layers, as illustrated in Figure 2(d). Figure 7 plots the final point sets obtained by the five archivers on the sequence. As can be seen, the decomposition-based archivers have shown very good performance, with their points being distributed excellently over the front. The reason for this is that the "global" ideal point used in the calculation of the scalarising functions for the decomposition-based archivers can be found at the beginning of the archiving process (i.e. the minimum on each objective comes from the points fed to the archive at the early stage of the archiving). This will lead to the weights to spread perfectly over the whole triangular hyperplane.

In contrast, the two indicator-based archivers appear to struggle, especially IBEA whose final points are around the three medians of the triangle. To understand how this unusual distribution pattern appears, we plot the archive obtained by IBEA at intermediate iterations 1000th, 2000th,



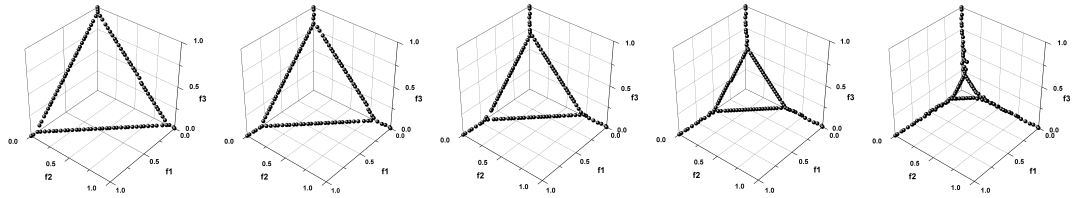
(a) NSGA-II HVR=92.207% (b) IBEA HVR=89.349% (c) SMS-EMOA HVR=89.506% (d) MOEA/D HVR=77.017% (e) NSGA-III HVR=86.480%

Fig. 6. The final point sets obtained by the five archiving methods on Sequence 3 and their corresponding HVR results.



(a) NSGA-II HVR=92.311% (b) IBEA HVR=83.717% (c) SMS-EMOA HVR=95.092% (d) MOEA/D HVR=96.517% (e) NSGA-III HVR=96.681%

Fig. 7. The final point sets obtained by the five archiving methods on Sequence 4 and their corresponding HVR results. Note that all the points are on the simplex, including those obtained by the IBEA archiver, which are located around the three medians of the simplex.



(a) Iter. 1000th (b) Iter. 2000th (c) Iter. 3000th (d) Iter. 4000th (e) Iter. 5000th

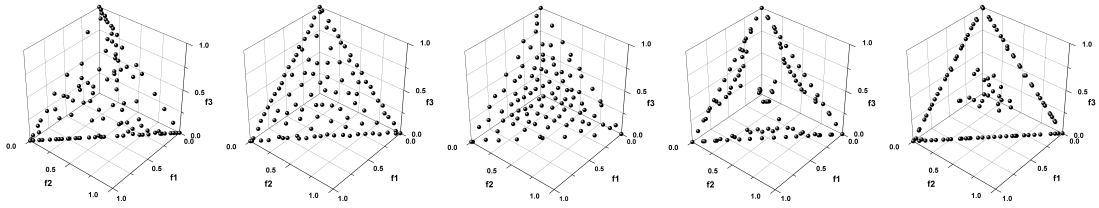
Fig. 8. The point set obtained by the IBEA archiver at intermediate iterations 1000th, 2000th, 3000th, 4000th and 5000th on Sequence 4.

3000th, 4000th and 5000th in Figure 8. As can be seen in the figure, with more inner-triangle-layer points arriving, only extreme points of a layer are preserved in the archive. In fact, the behaviour that IBEA preserves only extreme points of the Pareto front has been observed in the literature [40]. This eventually leads to the archive ending up with a set of points around the medians of the first outer triangle.

5.5 Sequence 5 (Figure 9)

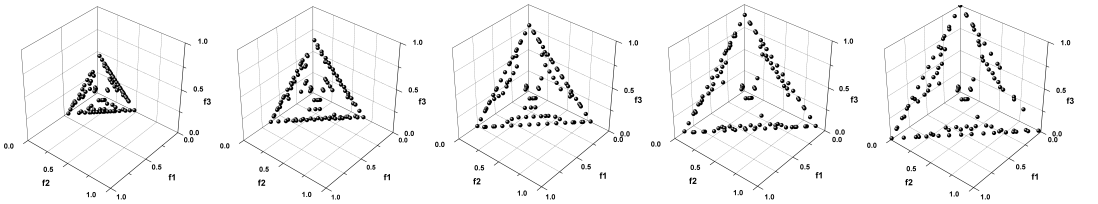
Exactly opposite to Sequence 4, the order of the points in Sequence 5 is from inner layers to outer layers, as shown in Figure 2(e). Unsurprisingly, their behaviours are rather different from those for Sequence 4. Figure 9 plots the final point sets obtained by the five archivers on Sequence 5. As seen from the figure, the biggest contrast is from IBEA whose archive is maintained pretty well. This occurrence can be attributed to the fact that the extreme points of the inner triangle layers, with new outer-layer points arriving, are not extreme ones any more (as they are inside of the outer triangle layer), thus not preferred by IBEA. In addition, interestingly, the point set obtained by NSGA-II is affected by the order of the points, which is not the case for its 2D version (i.e., Sequence 3). One possible explanation for this is that the crowding distance used in NSGA-II is inaccurate to measure points' crowdedness when the dimension is up to three [35, 41].

As for the decomposition-based archivers in MOEA/D and NSGA-III, unlike the situation on Sequence 4, they fail to return well-distributed sets on the sequence. Their final solutions concentrate in the boundary of the triangle, with only several inner solutions. The reason for this is that the ideal point, which is used in the calculation of the scalarising function in the decomposition-based archivers, keeps changing with the introduction of new points fed. Since the points arrive from



(a) NSGA-II HVR=95.120% (b) IBEA HVR=96.251% (c) SMS-EMOA HVR=96.267% (d) MOEA/D HVR=94.797% (e) NSGA-III HVR=94.600%

Fig. 9. The final point sets obtained by the five archiving methods on Sequence 5 and their corresponding HVR results.



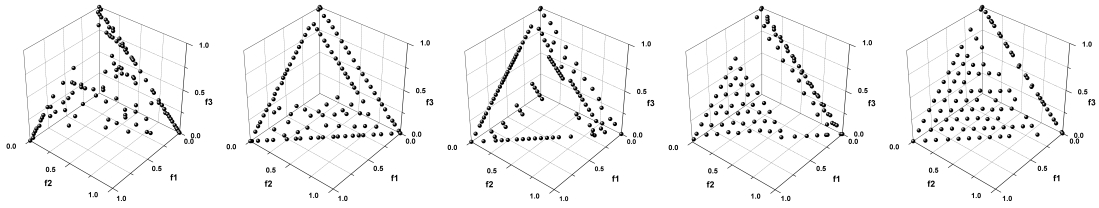
(a) Iter. 1000th (b) Iter. 2000th (c) Iter. 3000th (d) Iter. 4000th (e) Iter. 5000th

Fig. 10. The point set obtained by the MOEA/D archiver at intermediate iterations 1000th, 2000th, 3000th, 4000th and 5000th on Sequence 5.

inner triangle layers to outer triangle layers, the boundary solutions of the old triangles become inner solutions of the new triangles. For a current boundary solution, it is very likely for the archive to keep it since it may achieve the best scalarising function value on a boundary weight vector. For an inner solution, as essentially the change of the ideal point means the change of the weight directions, it may not always associate with one weight vector at every update step during the archiving process. In other words, with the constant change of weight directions, it is highly unlikely that at every iteration there always exists one weight vector on which that same inner solution achieves the best scalarising function value. Therefore, inner solutions are often eliminated by the new boundary solutions. Figure 10 shows the point set obtained by the MOEA/D archiver at several intermediate iterations. As seen, most of the boundary solutions of the triangle are kept. In contrast, very few inner solutions are kept, and the number of inner solutions decreases during the archiving process.

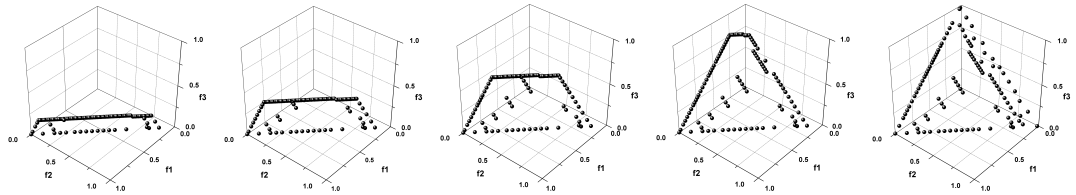
5.6 Sequence 6 (Figure 11)

As shown in Figure 2(f), the points in Sequence 6 arrive sequentially in a zigzag pattern, and with a turn in the middle. On this sequence, all the five archivers appear to struggle (Figure 11). The two decomposition-based archivers only maintain well the bottom-left triangle where its global ideal point can be found at the beginning of the archiving process. An interesting result is from SMS-EMOA which manages "okay" in the last two cases, but performs poorly here. To understand how this happened, we plot the archive maintained by SMS-EMOA at intermediate iterations 1000th, 2000th, 3000th, 4000th and 5000th in Figure 12. As can be seen, when archiving the bottom-left trapezoid-shape points, SMS-EMOA prefers boundary points, particularly those fed very recently. This is because that the hypervolume-based criterion may fail to maintain a point set with the



(a) NSGA-II HVR=92.880% (b) IBEA HVR=95.338% (c) SMS-EMOA HVR=95.012% (d) MOEA/D HVR=93.647% (e) NSGA-III HVR=96.011%

Fig. 11. The final point sets obtained by the five archiving methods on Sequence 6 and their corresponding HVR results.



(a) Iter. 1000th (b) Iter. 2000th (c) Iter. 3000th (d) Iter. 4000th (e) Iter. 5000th

Fig. 12. The point set obtained by the SMS-EMOA archiver at intermediate iterations 1000th, 2000th, 3000th, 4000th and 5000th on Sequence 6.

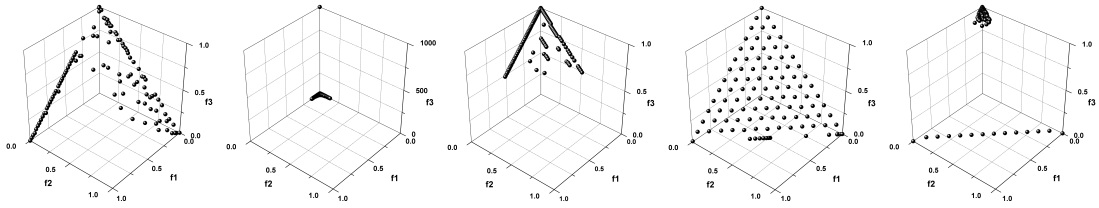
non-simplex shape in the 3D (or higher) space, especially those with many points sharing the same maximum value on an objective, as shown in [24, 25]. When the reference point is not set properly, the boundary points contribute much more than the inner ones to the hypervolume metric.

5.7 Sequence 7 (Figure 13)

Sequence 7, with a DRS point appearing at the beginning of the archiving process (see Figure 2(g)), seems the hardest one for the archivers except MOEA/D to maintain. As can be seen from Figure 13, IBEA cannot eliminate the DRS point and only keeps very late points in the sequence, and the points obtained by NSGA-II, SMS-EMOA and NSGA-III concentrate around some boundaries of the triangle. An interesting observation is that the decomposition-based MOEA/D, which shares a similar pattern to NSGA-III in the previous three 3D sequences, performs very well on the sequence. The reason for this is that the generation of the weight vectors in MOEA/D is not affected by the DRS points, but only the best value on each objective. Once the global ideal point for determining a set of optimal weight directions can be found at the beginning of the archiving process, as in this sequence, the points can be maintained very well. In contrast, in NSGA-III a normalisation operation is needed in order to generate the weight vectors, where both the best and worst value of all nondominated solutions on each objective is considered. The DRS point, which is a nondominated solution but performs extremely poorly on some objective, massively affects the accuracy of the normalisation in NSGA-III.

5.8 Results of MGA

The above results have shown the struggle of three classes of the five representative and widely used archivers in maintaining a set of points on simple simplex Pareto front shapes. One may ask what



(a) NSGA-II HVR=89.649% (b) IBEA HVR=22.569% (c) SMS-EMOA HVR=60.165% (d) MOEA/D HVR=95.958% (e) NSGA-III HVR=73.318%

Fig. 13. The final point sets obtained by the five archiving methods on Sequence 7 and their corresponding HVR results.

about more advanced archivers, particularly those with desirable theoretical properties[5]. In this section, we consider a state-of-the-art archiver, multi-level grid archiver (MGA) [38], which satisfies several desirable properties such as $\lhd$-monotone, limit-stable and limit-optimal [47]. MGA has polynomial computational complexity in the size of the solution set and the number of objectives, and it has been demonstrated to work very well in many scenarios, such as on the sequences presented in [47].

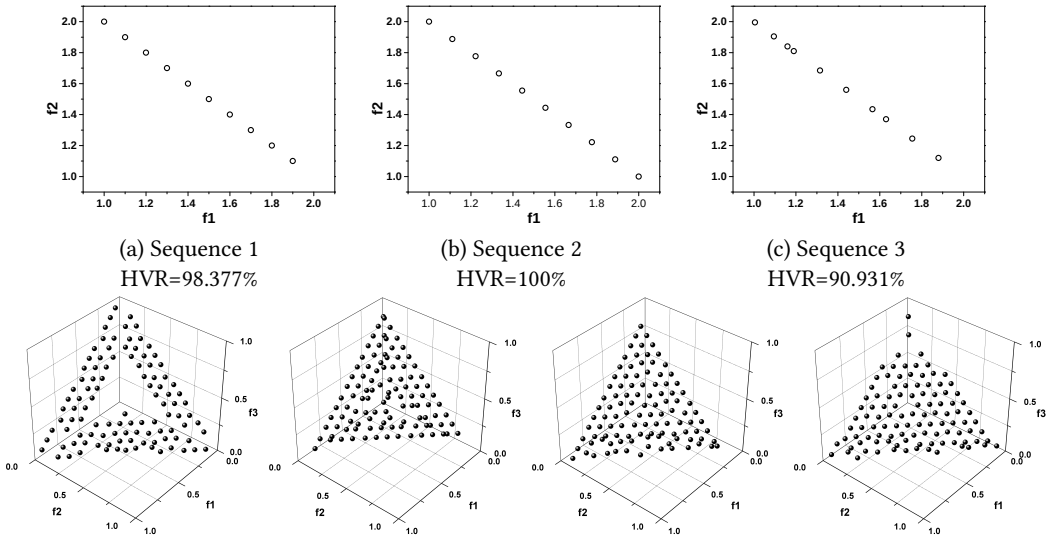
Figure 14 shows the final point sets obtained by MGA on the seven sequences. As can be seen, the results are mixed. On some sequences, MGA performs very well. For example, it can maintain all the Pareto optimal solutions on Sequence 2. Yet on some other sequences, MGA performs poorly. For example, on the four tri-objective sequences, MGA misses all the three extreme points of the simplex, and also some part of the Pareto front, namely, some inner layers of Sequence 4, outer layers of Sequence 5, right part of Sequence 6, and upper part of Sequence 7. Some possible explanations for this occurrence are as follows. Firstly, the grid/ϵ dominance criteria struggle to keep the extreme and boundary solutions due to the relaxation of the Pareto dominance (i.e., boundary solutions are likely to be dominated by inner ones). This has been reported in various grid/ϵ-based EMO algorithms [12, 21, 59]. Secondly, when the archive is at the maximum capacity and all of its solutions are at the same grid level (i.e., each in its own box), the archive may not accept a new solution with the same/finer grid level. That is why MGA maintains well the early part (i.e., early-coming points) of Sequences 4–7, but fail to keep the late part.

5.9 Summary

Based on the observations made in the previous sections, the following comments with respect to the three classes of archiving methods as well as MGA can be drawn.

- Archivers that consider Pareto dominance and density to distinguish between solutions (e.g., in NSGA-II) appear to suffer from the deterioration more than the other two classes of archivers. This is consistent with the observations in [43]. However, Pareto-based archivers are little affected by the case that the points fed from one region to another region along the Pareto front (e.g., shrinking and expanding), particularly for the 2D case.
- For indicator-based archivers, the points arriving in a shrinking pattern appear to be more challenging than those in an expanding pattern. Such archivers maintain the points well when they are in the shape of a filled triangle (e.g. on Sequence 5).

[5]The archiver in SMS-EMOA, despite using the hypervolume criterion, may not hold desirable theoretical properties since its reference point is adaptively updated during the archiving process.



(a) Sequence 1 HVR=98.377% (b) Sequence 2 HVR=100% (c) Sequence 3 HVR=90.931%

(d) Sequence 4 HVR=94.346% (e) Sequence 5 HVR=84.301% (f) Sequence 6 HVR=85.617% (g) Sequence 7 HVR=89.662%

Fig. 14. The final point sets obtained by MGA on the seven sequences and their corresponding HVR results.

- Opposite to indicator-based archivers, decomposition-based ones can handle the shrinking pattern very well, but they fail on the expanding pattern where the ideal point used in the calculation of the scalarising function keeps changing.
- It is known that Pareto-based archivers are difficult to get rid of the DRS points [23], but, interestingly, we here have shown that they can also pose a big challenge to many other archivers. Three of the four indicator/decomposition-based archivers struggle when the sequence has a DRS point, no matter whether the normalisation operation is conducted (e.g., in IBEA and NSGA-III) or not (e.g., in SMS-EMOA). MOEA/D is the only algorithm which can get rid of the DRS point in all the cases (Sequences 2 and 7), but still some small region on the Pareto front is not well covered.
- Archivers with desirable theoretical properties like MGA may not necessarily produce better results than those without. This is not very surprising since the theoretical properties are concerned with convergence (e.g., deterioration); here, however, all the sequences except Sequence 1 are introduced to challenge archivers in maintaining the diversity (i.e., spread and uniformity) of the solutions over the Pareto front.

6 DISCUSSIONS

With very regular Pareto front shapes, our experiments have shown the struggle of popular archiving methods in maintaining solution sets. This may explain the observations reported recently in [56] of why there is a big difference of the quality between the final population/archive and all the nondominated solutions found during the search. One straightforward way to deal with this unwelcome issue is to design new archiving methods more robust to the order of the solutions produced. Another solution is to develop new search models of EMO algorithms that fit the behaviour of the archiver used. For example, when a decomposition-based archiver is used, the

search for the best value on each objective of the MOP (in order to determine the global ideal point used in the scalarising function) could be conducted first.

On the other hand, one may consider to use an unbounded archive to store all the nondominated solutions found during the search. This has been practised in many studies such as [16, 19, 26, 34, 57, 58]. With the power of today's computers and efficient data structures [20, 28], this approach may become increasingly feasible.

It is necessary to point out that the observations reported here are derived from the (μ+1) archiving mechanism, which may differ from the (μ+μ) archiving mechanism. Yet, which mechanism produces better results may depend on specific archiving methods used in EMO algorithms. Based on whether or not to adjust the "fitness" of solutions (i.e., the evaluation values of solutions under the archiving criterion) during the archiving process, there are two types of methods in the ($\mu + \mu$) mechanism in the area. One is to calculate the fitness of all the $\mu + \mu$ solutions and then remove the worst μ solutions at once, i.e., without adjusting the remaining solutions' fitness after any solution has been removed. Such methods include the crowding distance-based method in NSGA-II. These types of ($\mu + \mu$) archiving method are likely to lead to worse results than their ($\mu + 1$) counterparts since a solution's fitness is based on the comparison with other solutions in the considered set and thus is changed when the set changes. This is actually the reason that in the experiments the NSGA-II archiver (with the ($\mu + 1$) archiving mechanism) performs very well on Sequence 3. The other type of ($\mu + \mu$) archiving methods is to remove the μ solutions one by one, i.e., find the worst solution in the considered set, remove it, and adjust the fitness of the remaining solutions; these steps are repeated until μ solutions are removed. Such methods include the archiving method in IBEA. This type of archiving methods is likely to produce better results than their ($\mu + 1$) counterparts since it is like conducting the ($\mu + 1$) archiving but knowing some future inputs.

Lastly, it is worth mentioning that in the evolutionary search, the population replacement/update process can be different from the archive truncation process despite the fact that they can share the same selection criteria/rules. The former typically comes with a fixed-size population; dominated solutions can also be preserved provided that there are some slots in the population which have not been filled by nondominated ones. Preserving such dominated solutions is conducive since it can lead to lower chance of premature convergence than maintaining a smaller population which only consists of nondominated solutions [6, 39]. Fortunately, the results obtained by this work can also apply to population replacement/update since the new point fed to the archive is always nondominated to the current archive on all the sequences.

7 CONCLUSION

Solution generation and maintenance are two cornerstones that make up an EMO algorithm, particularly for the latter (aka archiving) which is the focus of studies in the area over decades. Some algorithms, in essence, only design solution maintenance rules, such as NSGA-III — their solution generation component (mating selection, crossover and mutation) follows common practice. It is well known that the effectiveness of archiving methods is significantly affected by the shape of problems' Pareto front, but it is long believed that archiving methods work well on simple simplex Pareto fronts provided that all the solutions are available (i.e., can be found by the search algorithm).

In this study, we challenged this belief by showing that the order of solutions produced matters. We constructed seven solution sequences whose Pareto front is a 1D or 2D simplex. We showed that the struggle of five representative archiving methods as well as a more advanced one — none of them performs well on half of the sequences (Table 3). This suggests archiving techniques still far from maturity and new methodologies needed.

The complexity and variety of real-world optimisation problems and randomness elements in evolutionary search can in theory lead to any solution sequence. As there is no knowledge of future

inputs, it is difficult for archiving methods to decide which solutions to preserve and which to discard, as we have shown. It may be worth studying the archiving operation more independently (i.e., separating from the design of search algorithms), e.g., by considering solution sequences having a variety of orders, shapes, etc. This would avoid interference from randomness elements involved in the evolutionary search, and help quickly detect problems of archiving methods through various sequences that could be easily constructed.

REFERENCES

[1] Hernán Aguirre, Saúl Zapotecas, Arnaud Liefooghe, Sébastien Verel, and Kiyoshi Tanaka. 2016. Approaches for many-objective optimization: Analysis and comparison on MNK-landscapes. In *International Conference on Artificial Evolution*. Springer, 14–28.

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

[3] Leonardo CT Bezerra, Manuel López-Ibánez, and Thomas Stützle. 2017. An empirical assessment of the properties of inverted generational distance on multi-and many-objective optimization. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 31–45.

[4] Leonardo CT 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*. 620–628.

[5] Leonardo CT Bezerra, Manuel López-Ibáñez, and Thomas Stützle. 2020. Automatic configuration of multi-objective optimizers and multi-objective configuration. In *High-Performance Simulation-Based Optimization*. Springer, 69–92.

[6] Leonardo CT Bezerra, Manuel López-Ibáñez, and Thomas Stützle. 2020. Automatically designing state-of-the-art multi-and many-objective evolutionary algorithms. *Evolutionary computation* 28, 2 (2020), 195–226.

[7] Karl Bringmann and Tobias Friedrich. 2014. Convergence of hypervolume-based archiving algorithms. *IEEE Transactions on Evolutionary Computation* 18, 5 (2014), 643–657.

[8] Tao Chen, Miqing Li, and Xin Yao. 2019. Standing on the shoulders of giants: Seeding search-based multi-objective optimization with prior knowledge for software service composition. *Information and Software Technology* 114 (2019), 155–175.

[9] Carlos A Coello Coello and Margarita Reyes Sierra. 2004. A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. In *Proceedings of the Mexican International Conference on Artificial Intelligence (MICAI)*. 688–697.

[10] Constantinos Daskalakis, Ilias Diakonikolas, and Mihalis Yannakakis. 2016. How good is the Chord algorithm? *SIAM J. Comput.* 45, 3 (2016), 811–858.

[11] Kalyanmoy Deb and Himanshu 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] Kalyanmoy Deb, Manikanth Mohan, and Shikhar Mishra. 2005. Evaluating the ϵ-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions. *Evolutionary Computation* 13, 4 (Dec. 2005), 501–525.

[13] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197.

[14] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. 2005. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, Ajith Abraham, Lakhmi Jain, and Robert Goldberg (Eds.). Berlin, Germany: Springer, 105–145.

[15] Jérémie Dubois-Lacoste, Manuel López-Ibáñez, and Thomas Stützle. 2011. Improving the anytime behavior of two-phase local search. *Annals of Mathematics and Artificial Intelligence* 61, 2 (2011), 125–154.

[16] Richard M Everson, Jonathan E Fieldsend, and Sameer Singh. 2002. Full elite sets for multi-objective optimisation. In *Adaptive Computing in Design and Manufacture V*. Springer, 343–354.

[17] Jesús Guillermo Falcón-Cardona and Carlos A Coello Coello. 2020. Indicator-based multi-objective evolutionary algorithms: a comprehensive survey. *Comput. Surveys* 53, 2 (2020), 1–35.

[18] Jonathan E. Fieldsend. 2017. University staff teaching allocation: formulating and optimising a many-objective problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*. ACM, 1097–1104.

[19] Jonathan E. Fieldsend, Richard M. Everson, and Sameer Singh. 2003. Using unconstrained elite archives for multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 7, 3 (2003), 305–323.

[20] Tobias Glasmachers. 2017. A fast incremental BSP tree archive for non-dominated points. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 252–266.

[21] David Hadka and Patrick Reed. 2013. Borg: An auto-adaptive many-objective evolutionary computing framework. *Evolutionary Computation* 21, 2 (2013), 231–259.
[22] Zhenan He and Gary Yen. 2016. Many-objective evolutionary algorithm: Objective space reduction + diversity improvement. *IEEE Transactions on Evolutionary Computation* 20, 1 (2016), 145–160.
[23] K. Ikeda, H. Kita, and S. Kobayashi. 2001. Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal?. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, Vol. 2. 957–962.
[24] Hisao. Ishibuchi, Ryo Imada, Yu Setoguchi, and Yusuke Nojima. 2017. Reference point specification in hypervolume calculation for fair comparison and efficient search. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 585–592.
[25] Hisao. Ishibuchi, Ryo Imada, Yu Setoguchi, and Yusuke Nojima. 2018. How to specify a reference point in hypervolume calculation for fair performance comparison. *Evolutionary Computation* 26, 3 (2018), 411–440.
[26] Hisao Ishibuchi, Lie Meng Pang, and Ke Shang. 2020. A new framework of evolutionary multi-objective algorithms with an unbounded external archive. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*.
[27] Hisao Ishibuchi, Yu Setoguchi, Hiroyuki Masuda, and Yusuke Nojima. 2017. Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. *IEEE Transactions on Evolutionary Computation* 21, 2 (2017), 169–190.
[28] Andrzej Jaszkiewicz and Thibaut Lust. 2018. ND-tree-based update: a fast algorithm for the dynamic nondominance problem. *IEEE Transactions on Evolutionary Computation* 22, 5 (2018), 778–791.
[29] Siwei Jiang, Y-S Ong, Jie Zhang, and Liang Feng. 2014. Consistencies and contradictions of performance metrics in multiobjective optimization. *IEEE Transactions on Cybernetics* 44, 12 (2014), 2391–2404.
[30] Joshua Knowles and David Corne. 2004. Bounded Pareto archiving: Theory and practice. In *Metaheuristics for multiobjective optimisation*. Springer, 39–64.
[31] Joshua D Knowles. 2002. *Local-search and hybrid evolutionary algorithms for Pareto optimization.* Ph.D. Dissertation. University of Reading UK.
[32] Joshua D. Knowles and David W. Corne. 2003. Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Transactions on Evolutionary Computation* 7, 2 (2003), 100–116.
[33] Joshua D. Knowles, David W. Corne, and Mark Fleischer. 2003. Bounded archiving using the Lebesgue measure. In *Congress on Evolutionary Computation (CEC)*, Vol. 4. IEEE, 2490–2497.
[34] Oswin Krause, Tobias Glasmachers, Nikolaus Hansen, and Christian Igel. 2016. Unbounded population MO-CMA-ES for the bi-objective BBOB test suite. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*. 1177–1184.
[35] S. Kukkonen and K. Deb. 2006. A fast and effective method for pruning of non-dominated solutions in many-objective problems. In *Parallel Problem Solving from Nature (PPSN IX)*. Reykjavik, Iceland, 553–562.
[36] Frank Kursawe. 1990. A variant of evolution strategies for vector optimization. In *International Conference on Parallel Problem Solving from Nature*. Springer, 193–197.
[37] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. 2002. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation* 10, 3 (2002), 263–282.
[38] Marco Laumanns and Rico Zenklusen. 2011. Stochastic convergence of random search methods to fixed size Pareto front approximations. *European Journal of Operational Research* 213, 2 (2011), 414–421.
[39] Ke Li, Renzhi Chen, Guangtao Fu, and Xin Yao. 2018. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 23, 2 (2018), 303–315.
[40] Miqing Li, Crina Grosan, Shengxiang Yang, Xiaohui Liu, and Xin Yao. 2018. Multi-line distance minimization: A visualized many-objective test problem suite. *IEEE Transactions on Evolutionary Computation* 22, 1 (2018), 61–78.
[41] Miqing Li, Shengxiang Yang, and Xiaohui Liu. 2014. Shift-based density estimation for Pareto-based algorithms in many-objective optimization. *IEEE Transactions on Evolutionary Computation* 18, 3 (2014), 348–365.
[42] Miqing Li, Shengxiang Yang, and Xiaohui Liu. 2016. Pareto or non-Pareto: Bi-criterion evolution in multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 20, 5 (2016), 645–665.
[43] Miqing Li and Xin Yao. 2019. An empirical investigation of the optimality and monotonicity properties of multiobjective archiving methods. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 15–26.
[44] Miqing Li and Xin Yao. 2019. Quality evaluation of solution sets in multiobjective optimisation: A survey. *Comput. Surveys* 52, 2 (2019).
[45] Miqing Li and Xin Yao. 2020. What weights work for you? Adapting weights for any Pareto front shape in decomposition-based evolutionary multiobjective optimisation. *Evolutionary Computation* 28, 2 (2020), 227–253.
[46] Hai-Lin Liu, Fangqing Gu, and Qingfu Zhang. 2013. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Transactions on Evolutionary Computation* 18, 3 (2013), 450–455.
[47] Manuel López-Ibáñez, Joshua Knowles, and Marco Laumanns. 2011. On sequential online archiving of objective vectors. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 46–60.

[48] Manuel López-Ibáñez, Arnaud Liefooghe, and Sébastien Verel. 2014. Local optimal sets and bounded archiving on multi-objective NK-landscapes with correlated objectives. In *International Conference on Parallel Problem Solving from Nature*. Springer, 621–630.

[49] Thibaut Lust and Jacques Teghem. 2010. Two-phase Pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics* 16, 3 (2010), 475–510.

[50] Luis Paquete and Thomas Stützle. 2003. A two-phase local search for the biobjective traveling salesman problem. In *International Conference on Evolutionary Multi-Criterion Optimization*. 479–493.

[51] R. C. Purshouse and P. J. Fleming. 2007. On the Evolutionary Optimization of Many Conflicting Objectives. *IEEE Transactions on Evolutionary Computation* 11, 6 (2007), 770–784.

[52] Günter Rudolph and Alexandru Agapie. 2000. Convergence properties of some multi-objective evolutionary algorithms. In *Proceedings of the 2000 Congress on Evolutionary Computation*, Vol. 2. IEEE, 1010–1016.

[53] Günter Rudolph, Oliver Schütze, Christian Grimme, Christian Domínguez-Medina, and Heike Trautmann. 2016. Optimal averaged Hausdorff archives for bi-objective problems: theoretical and numerical results. *Computational Optimization and Applications* 64, 2 (2016), 589–618.

[54] Oliver Schütze, Marco Laumanns, Carlos A Coello Coello, Michael Dellnitz, and El-Ghazali Talbi. 2008. Convergence of stochastic search algorithms to finite size Pareto set approximations. *Journal of Global Optimization* 41, 4 (2008), 559–577.

[55] Sarah Steiner and Tomasz Radzik. 2008. Computing all efficient solutions of the biobjective minimum spanning tree problem. *Computers & Operations Research* 35, 1 (2008), 198–211.

[56] Ryoji Tanabe, Hisao Ishibuchi, and Akira Oyama. 2017. Benchmarking multi-and many-objective evolutionary algorithms under two optimization scenarios. *IEEE Access* 5 (2017), 19597–19619.

[57] Ryoji Tanabe and Akira Oyama. 2017. Benchmarking MOEAs for multi-and many-objective optimization using an unbounded external archive. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 633–640.

[58] Hao Wang, Chaoli Sun, Yaochu Jin, Shufen Qin, and Haibo Yu. 2019. A Multi-indicator based Selection Strategy for Evolutionary Many-objective Optimization. In *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2042–2049.

[59] Shengxiang Yang, Miqing Li, Xiaohui Liu, and Jinhua Zheng. 2013. A grid-based evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 17, 5 (2013), 721–736.

[60] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (2007), 712–731.

[61] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari. 2009. *Multiobjective optimization test instances for the CEC 2009 special session and competition.* Working Report CES-487. School of CS & EE, University of Essex.

[62] Eckart Zitzler and Simon Künzli. 2004. Indicator-based selection in multiobjective search. In *Proceedings of the International Conference on Parallel Problem Solving from Nature (PPSN)*. 832–842.

[63] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2002. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design, Optimisation and Control*. Barcelona, Spain, 95–100.

[64] Eckart Zitzler and Lothar Thiele. 1999. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation* 3, 4 (1999), 257–271.