Université Libre de Bruxelles
*Institut de Recherches Interdisciplinaires
et de Développements en Intelligence Artificielle*

On Sequential Online Archiving of Objective Vectors

Manuel López-Ibáñez, Joshua Knowles, and Marco Laumanns

IRIDIA – Technical Report Series

Technical Report No.
TR/IRIDIA/2011-001

January 2011
Last revision: March 2011

IRIDIA – Technical Report Series
ISSN 1781-3794

Published by:
IRIDIA, *Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle*
Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium

Technical report number TR/IRIDIA/2011-001

Revision history:
TR/IRIDIA/2011-001.002 March 2011

On Sequential Online Archiving of Objective Vectors

Manuel López-Ibáñez[1], Joshua Knowles[2], and Marco Laumanns[3]

[1] IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
manuel.lopez-ibanez@ulb.ac.be
[2] School of Computer Science, University of Manchester, UK
j.knowles@manchester.ac.uk
[3] IBM Research – Zurich, Switzerland
mlm@zurich.ibm.com

Abstract. In this paper, we examine the problem of maintaining an approximation of the set of nondominated points visited during a multiobjective optimization, a problem commonly known as archiving. Most of the currently available archiving algorithms are reviewed, and what is known about their convergence and approximation properties is summarized. The main scenario considered is the restricted case where the archive must be updated online as points are generated one by one, and at most a fixed number of points are to be stored in the archive at any one time. In this scenario, the $\lhd$-monotonicity of an archiving algorithm is proposed as a weaker, but more practical, property than negative efficiency preservation. This paper shows that hypervolume-based archivers and a recently proposed multi-level grid archiver have this property. On the other hand, the archiving methods used by SPEA2 and NSGA-II do not, and they may $\lhd$-deteriorate with time. The $\lhd$-monotonicity property has meaning on any input sequence of points. We also classify archivers according to limit properties, i.e. convergence and approximation properties of the archiver in the limit of infinite (input) samples from a finite space with strictly positive generation probabilities for all points. This paper establishes a number of research questions, and provides the initial framework and analysis for answering them.

Keywords: approximation set, archive, convergence, efficiency preserving, epsilon-dominance, hypervolume, online algorithms

1 Introduction

The convergence properties of large classes of multiobjective evolutionary algorithms were seriously considered for the first time in the late 1990s [18,9,17]. These papers laid the foundations for much of the analysis that has gone on to date, and showed that certain types of elitism combined with a certain type of generation process lead to convergence (in the limit) to a subset of the Pareto front (PF). Moreover, they indicated that, to a large extent, properties of a multiobjective stochastic search algorithm as a whole can be derived from separately considering properties of the generation process and properties of the elite-preserving mechanism.

Today, most multiobjective stochastic search algorithms are elitist in the sense of keeping an external *archive* (or memory) in order to capture the output of the search process. Because the set of minima visited may be very large in a multiobjective optimization process, it is common to bound the size of the archive. Thus, properties of the elite-preservation, or archiving, rules used to maintain bounded archives are of high interest to the community. Our aim in this paper is to elucidate, in one place, some of the properties of existing archiving algorithms that keep at most a fixed maximum number of points to approximate the PF. We restrict our attention to sequential archiving of points that arrive one-by-one, but consider a number of differently motivated algorithms for this setting. We consider archiving algorithms aimed only at convergence (similar to AR1 [17]), algorithms aimed mostly at 'diversity'[4] (derived from the elite population update rules of SPEA2 [19] and NSGA-II [7]), algorithms that consider overall approximation quality (epsilon dominance-based [16], grid-based [13], and based on maximizing hypervolume [11,1]), including a relatively new proposal called multi-level grid archiving [15]. We review the properties of these archiving algorithms and illustrate them empirically.

2 Preliminaries

We are concerned with vectors (points) in finite, multidimensional objective spaces. Let $Y \subset \mathbb{R}^d$ be a finite objective space of dimension $d > 1$. An order relation on Y may be defined as follows: $y \prec y'$ iff $\forall i \in 1, \ldots, d,\ y_i \leq y'_i$ and $y \neq y'$. Thus $\prec$ is a strict partial order on Y. Instead of $y \prec y'$ we may also write y *dominates* y'. The set of minimal elements of Y may be defined as

$$Y^* := \min(Y, \prec) = \{y \in Y, \nexists y' \in Y, y' \prec y\}.$$

The set Y^* is called the *Pareto front* (PF). Any other set $P \subseteq Y$ with the property $P = \min(P, \prec)$ will be called a *nondominated set*.

We are interested in finding approximations of the set Y^* of cardinality at most N. Such approximation sets are also partially ordered when we extend the definitions of dominance to pairs of sets as follows. Let P be a nondominated set. A point $y \notin P$ is *nondominated* w.r.t. P iff $\nexists y' \in P, y' \prec y$. Let P and Q be two nondominated sets. Then $P \lhd Q$ iff $\min(P \cup Q, \prec) = P \neq Q$.

2.1 Optimal approximation sets of bounded size

The partial order on sets defined by $\lhd$ gives the primary *solution concept* for determining an optimal approximation set of size at most N, as follows:

Definition 1 (Optimal Approximation Set of Bounded Size). *If $A \subseteq Y$ is a nondominated set, $|A| \leq N$, and $\nexists B \subseteq Y, |B| \leq N, B \lhd A$, then A is an* optimal approximation set of bounded size *N of Y^*.*

This solution concept derives from the dominance partial order only, but is in general not sufficient to guide a search or archiving process alone. We are now used to the notion of evaluating approximation sets with performance indicators, and using performance indicators to define other solution concepts that are *compatible* with dominance (see below), i.e. they are *refinements* [21] of it, that may be more suitable for guiding an archiving algorithm.

2.2 Compatibility of performance indicators

Let J be the set of all nondominated subsets of Y. A *unary performance indicator* $I : J \to \mathbb{R}$ is a mapping from the set J to the real numbers. Assuming that the indicator's value is to be minimised, we can define compatibility of I with respect to $(J, \lhd)$. If $\nexists A, B \in J$, such that $A \lhd B$ and $I(A) \geq I(B)$ then I is a *compatible* indicator [10,22]. Analogously, a *weakly compatible indicator* can be defined by replacing $I(A) \geq I(B)$ with $I(A) > I(B)$ in the statement above.

[4] The term "diversity" has no fixed definition in the literature, but it can refer to the evenness of the spacing between points and/or the extent of the nondominated set.

Hypervolume indicator The hypervolume indicator HYP(A) [20] of an approximation set A (originally called the $\mathcal{S}$ metric in the literature) is the Lebesgue integral of the union of (hyperectangular, axis-parallel) regions dominated by the set A and bounded by a single d dimensional reference point that must be dominated by all members of the true PF. The indicator's value should be maximized. The compatibility of the indicator [12,22] is behind its importance as a performance assessment method and as a means of guiding search and archiving algorithms.

Additive ϵ indicator A point y is said to be weakly ϵ-dominated by a point y' iff $\forall i \in 1,\ldots,d$, $y'_i \leq y_i + \epsilon_i$. The unary epsilon indicator $\epsilon_{add}(A)$ of an approximation set A is defined as the minimum value of ϵ such that every point in Y^* is weakly ϵ-dominated by an element of A. This indicator has been shown to be weakly compatible with the $\triangleleft$-relation on sets [22] following the proposal of ϵ-dominance as a means of evaluating and obtaining approximation sets [16].

3 Archivers, Convergence and Approximation

Similarly to earlier papers [16,13,6], the setting we consider is that some generation process is producing a *sequence* of points (objective vectors) $\langle y^{(1)}, y^{(2)}, \ldots \rangle$, and we wish to maintain a subset of these minima in an archive A of fixed maximum size, $|A| \leq N$. We denote by A_t the contents of the archive after the presentation of the t-th objective vector. An *archiver*, i.e., an archiving algorithm for updating A with y, is an online algorithm [2] as it has to deal with a stream of data with no knowledge of future inputs. Knowles and Corne [13] previously showed that this online nature of the task means that no archiver can guarantee to have in its archive $\min(N, |Y^*_t|)$ where Y^*_t is the set of minimal elements of the input sequence up to a time t. A corollary of this, not previously stated explicitly, is that no online archiver of bounded size can deliver an 'optimal approximation set of bounded size' even in the weak sense of Definition 1.

3.1 Convergence and Approximation Definitions

When analysing an archiver's behaviour, we may be interested in how it performs in general input sequences of finite length, where points do not necessarily appear more than once in the sequence. This scenario models a one-pass finite sample of the search space. Or we may be interested in sequences where every point is seen an infinite number of times [17]. When considering the one-pass setting, we wish to know whether the archive is always a good approximation of the input sequence (at every time step). When considering the behaviour on points drawn indefinitely from a finite space, we wish to know whether convergence ever occurs (does the archive stop changing eventually?), and if so, what kind of approximation set is obtained, i.e. what is the archiver's limit behaviour. The following definitions expand on these ideas. The first four are properties that apply to one-pass settings (which also imply they are limit properties, too). Two limit-behaviour definitions follow.

Definition 2 ($\subseteq Y^*$). *No point in the archive is dominated by a point in the input sequence:* $\forall t,\ \forall y \in A_t, y \in Y^*_t$.

Definition 3 (diversifies). *An archiver is efficiency preserving [9] when full, if* $\forall t,\ |A_t| = N$, $y \in A_{t+1}$ *iff* $\exists y' \in A_t, y \prec y'$. *In other words, the archiver, when full, cannot accept points outside of the region dominating the current archive, thus limiting the diversity of points in the archive. This also implies that the archiver is* negative efficiency preserving, *that is, the region dominated by the current archive is a subset of the region dominated by subsequent archives [9]. We say that an archiver* without *this property* diversifies *by discarding a nondominated point from the archive to accept the new one.*

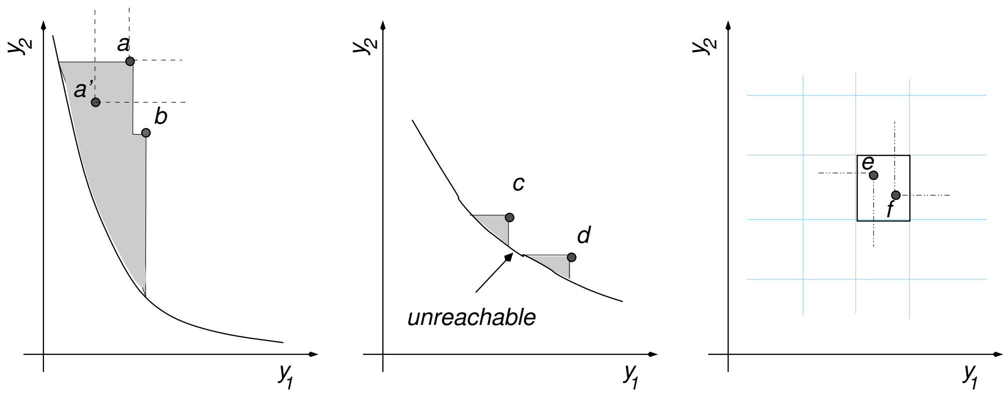


Fig. 1. Illustrations of some convergence concepts. (Left) Consider that $\{a, b\}$ is an archive; then an *efficiency-preserving* archiver may only accept a point in the (dominating) shaded region. If it accepts a' (removing a) this is also *negatively efficiency preserving* because the total region dominated is a superset of the region dominated by a, as indicated by the dashed lines. (Middle) Consider a different archive represented by c and d. In this case, a negatively efficiency preserving archiver can cause points to be unreachable, since only points within either of the shaded regions can now be accepted (adapted from Hanne [9]). (Right) Points e and f illustrate how the ϵ-Pareto archiver manages to guarantee that only Pareto optimal points are in its final archive. Two points in the same box cannot co-exist so one will be rejected from the archive. Let us say it is f. Only points which dominate e are allowed to populate the box in the future. Since the intersection of the region dominating e and the region dominated by f is empty, this ensures that, although f is no longer in the archive, no point dominated by f ever enters the archive.

Definition 4 (monotone). *There does not exist a pair of points $y \in A_t$ and $y' \in A_v$, $t < v$ such that y dominates y'. Let an archiver that does not have this property be said to* deteriorate*.*

Definition 5 ($\lhd$-monotone). *There does not exist a pair of sets A_t and A_v, $t < v$ such that $A_t \lhd A_v$. Let an archiver that does not have this property be said to* $\lhd$-deteriorate*.*

Definition 6 (limit-stable). *For any sequence consisting of points drawn indefinitely with a strictly positive probability from a finite set, there exists a t such that $\forall v > t$, $A_t = A_v$. That is, the archive set converges to a stable set in finite time.*

Definition 7 (limit-optimal). *For any sequence consisting of points drawn indefinitely with a strictly positive probability from a finite set, the archive will converge to an optimal bounded archive (see Definition 1).*

Table 1 summarises the properties of the eight archivers in terms of Definitions 2–7. An illustration of some of these concepts is given in Fig. 1.

3.2 Basic Archiver Pattern

Six of the eight archivers we study (all except for the two ϵ-based ones [16]) follow the scheme of Algorithm 1. These archivers describe a class called “precise” [6]. It is helpful for the later analysis of each individual archiver to observe the properties of Rules 1 and 2 (see Algorithm 1). Rule 1 is *efficiency-preserving* [9], which means that the region that contains points that dominate the archive after application of the rule is a subset of this region before the rule was applied. The rule is also *negative efficiency preserving* (Ibid.), which means that the region dominated by the

Table 1. Types of convergence behaviour displayed by the archivers, and broad indication of time complexity for archive update. P denotes polynomial in N and d, and $E(d)$ expresses exponential in d.

Archiver	$\subseteq Y^*$	Diversif.	Monotone	$\lhd$-monotone	Limit-stable	Limit-optimal	Complexity
Unbounded	+	+	+	+	+	+	P
Dominating	-	-	+	+	+	+	P
ϵ-approx	-	+	-	+	+	-	P
ϵ-Pareto	+	+	+	+	+	-	P
NSGA-II	-	+	-	-	-	-	P
SPEA2	-	+	-	-	-	-	P
AGA	-	+	-	-	-	-	P
AA_S	-	+	-	+	+	+	$E(d)$
MGA	-	+	-	+	+	+	P

Algorithm 1 Basic Archiver Pattern

Input: A_{t-1}, y
if $\exists y' \in A_{t-1}, y' \prec y$ **then**
  $A_t \leftarrow \min(A_{t-1} \cup \{y\})$ // Rule 1
else if $|\min(A_{t-1} \cup \{y\})| \leq N$ **then**
  $A_t \leftarrow \min(A_{t-1} \cup \{y\})$ // Rule 2
else
  $A_t \leftarrow \mathsf{filter}(A_{t-1} \cup \{y\})$ // $\mathsf{filter}(\cdot)$ returns a set of size N
end if
Output: A_t

archive after application of the rule is a superset of this region before. Rule 2 on the other hand is just *negative efficiency preserving*. For other properties of the algorithms described below, see Table 1.

3.3 Unbounded Archive

Trivially, the archiver yields the Pareto front of the input sequence. Although it is negative efficiency preserving [9] (Def. 3), it does not suffer from the curse of *unreachable points* (Ibid.) because these only occur when the set of points is also size limited.

3.4 Dominating Archive

The simplest way to achieve an archive of fixed maximum size is to implement the Basic Archiver with the $\mathsf{filter}(\cdot)$ function that just returns A_{t-1}. In other words, this archiver admits only dominating points whenever it is at full capacity. This archiver, when connected to a suitable sequence-generating process, is similar to the AR1 algorithm [17]. Due to the use of Rules 1 and 2 in combination (only), the archiver is negative efficiency preserving [9] (Def. 3). Two corollaries of this are that the archive cannot *deteriorate* (Def. 4), and it will always contain a subset of the Pareto front of the input sequence. However, the archiver gives no guarantee of approximation quality, and, in practice, especially for small N, it will tend to an almost *efficiency preserving* behaviour [9] where it shrinks into a small region of the Pareto front. The archive may also contain points that are not Pareto optimal in the input sequence (even though deterioration does not occur), because $|A|$ may fall below N (due to Rule 1) and points dominated in Y may be accepted because the dominating point in Y^* was previously rejected entry into the archive due to rule $\mathsf{filter}(\cdot)$, at an earlier timestep when the archive was full.

3.5 Adaptive ϵ-Approx Archiving

The ϵ-approx archiver [16] does not follow our previous pattern. In this algorithm, a point is accepted only if it is not ϵ-dominated by an archived point. If it is accepted, then $A_t \leftarrow \min(A_{t-1} \cup \{y\})$, as usual. For fixed ϵ, it was shown that the archive is always an ϵ-approximate set of the input sequence of finite size (but not limited to any fixed value).

Laumanns et al. [16] also describe an adaptive scheme in order to allow a user to specify a maximum archive size N, rather than an ϵ value. However, this scheme often results in too large values of ϵ with the result that too few points are archived (e.g. compared to AGA) [14]. Hence, although the archiver is $\lhd$-monotone, it is not limit-optimal. Other properties are summarised in Table 1.

3.6 Adaptive ϵ-Pareto Archiving

The second archiver in [16] uses the idea that objective space can be discretized, via ϵ, into equivalence classes called 'boxes', so that every objective vector belongs to precisely one box. Within a box, only one point is allowed to exist in the archive, and the update rule within a box allows only a dominating point to replace the incumbent (see Fig. 1). This scheme guarantees that every point in the archive is Pareto optimal wrt the input sequence. This is the only archiver here that has this property and maintains a size-bounded archive.

Similarly to the ϵ-approximate archiver, a scheme to adapt ϵ on the fly was also proposed in [16] so that an archive limited to N points could be obtained. But this adaptation scheme does not facilitate reducing ϵ if it starts or becomes too large, with the result that the archiver keeps too few solutions, preventing it from being limit-optimal.

3.7 NSGA-II Archiver

The NSGA-II algorithm [7] assigns different selective fitness to nondominated points on the basis of their *crowding distance*, a coarse estimate of the empty space that surrounds a point. Our NSGA-II archiver follows the scheme of the Basic Archiver (Algorithm 1), and implements the $\mathsf{filter}(\cdot)$ function by removing the point with minimum crowding distance [7].

Since crowding distance is independent of dominance, no convergence guarantees can be made. It does not yield a subset of the nondominated points from the input sequence, in general. More importantly, the archive may $\lhd$-deteriorate (Definition 5), and we later show this empirically in Section 4.4. Moreover, even on a sequence constructed from an indefinite random sampling of a finite space, the archive may never settle to a stable set.

3.8 SPEA2 Archiver

The external population update of SPEA2 [19] was designed to prevent some of the regression and oscillation observed in the original SPEA. Our SPEA2 archiver follows the scheme of the Basic Archiver (Algorithm 1), but uses the distance to the k-nearest neighbour as the density measure in the $\mathsf{filter}(\cdot)$ function, as is used in SPEA2 for update of the external population.

The SPEA2 archiver has similar properties to NSGA-II archiver in terms of convergence and approximation: The archive can $\lhd$-deteriorate, and the archiver is not limit-stable. Moreover, we show in Section 4.3 that even for a sequence of all Pareto-optimal points, the diversity measure of SPEA2 may lead to very poor approximation quality.

3.9 Adaptive Grid Archiving (AGA)

Adaptive grid archiving uses a grid over the points in objective space in order to estimate local density. Its $\mathsf{filter}(\cdot)$ rule in the instantiation of Algorithm 1 is

$$A_t \leftarrow A_{t-1} \cup \{y\} \setminus \{y_c \in C\}$$

where y_c is a point drawn uniformly at random from C, the set of all the vectors in the "most crowded" grid cells, excluding any points that are a minimum or maximum on any objective within the current archive.

The archive rule is neither negatively efficiency preserving nor nor avoids deterioration. Neither is it $\lhd$-monotone, a more serious problem. Only under special conditions (the grid cells are correctly sized and the grid stops moving) does a form of approximation guarantee become possible [11].

3.10 Hypervolume Archiver AA_S

This archiver was first proposed by Knowles [11] and follows the pattern of Algorithm 1, with the $\mathsf{filter}(\cdot)$ rule:

$$A_t \leftarrow \arg\max_{A \in \mathcal{A}_N} \{\mathrm{HYP}(A)\},$$

where $\mathcal{A}_N$ is the set of all subsets of $A_{t-1} \cup \{y\}$ of size N. In the one pass scenario, greedily removing the least-contributor does not ensure that the hypervolume is maximized over the whole sequence [4]. In Section 4.3, we provide an example where AA_S clearly does not maximize the hypervolume. Moreover, a point in the archive may be dominated by one that was previously in the archive, i.e., it may deteriorate. However, since the hypervolume never decreases, the archiver is $\lhd$-monotone (Definition 5).

The behaviour in the limit fulfills the solution concept (Definition 1), i.e. it is limit-optimal. The archive will be a set of $\min(N, |Y^*|)$ Pareto-optimal points after sufficiently long time, since if a set of size N has its maximum hypervolume value (out of all sets of such size) then all the points are Pareto optimal [8, Theorem 1].

Bringmann and Friedrich [5] have proved that hypervolume approximates the additive ϵ indicator, converging quickly as N increases. That is, sets that maximize hypervolume are near optimal on additive ϵ too, with the 'gap' diminishing as quickly as $O(1/N)$.

Updating the archive may be computationally expensive for large d and N. But despite the intractability of finding the point contributing least to the hypervolume in a set, approximation schemes may be good enough in practice [3].

3.11 Multi-level Grid Archiving (MGA)

The multi-level grid archiving (MGA) algorithm [15] can be thought of as combining principles from AGA and the ϵ-Pareto archiver. It was designed from the outset to maintain at most N points, achieving this by using a hierarchical family of boxes (equivalence classes) of different coarseness over the objective space. Specifically, when comparing solution at coarseness level $b \in \mathbb{Z}$, the components y_i of their objective vectors $y \in \mathbb{R}^d$ are mapped to (integral) values $\lfloor y_i \cdot 2^{-b} \rfloor$ to define its *box index vector* at level b.

The archiver follows the pattern of Algorithm 1. Its $\mathsf{filter}(\cdot)$ rule works by first determining the smallest level b where at least one of the $N+1$ candidates' box index vector is weakly dominated. The new candidate y is rejected if it belongs to the points that are weakly dominated at this level b; otherwise an arbitrary solution from this set is deleted. Through this adaptive determination of the right coarseness level for comparison, the behaviour observed in the ϵ-archivers of ending up with too large an ϵ value can be avoided, as we later show experimentally in Section 4.1.

The archiver may deteriorate, which means the archive may contain points dominated by points from previous archives. We provide an example of this in Section 4.5. Nevertheless, it is shown in [15] that any archive update strictly increases a unary performance indicator compatible with dominance, i.e., it is $\lhd$-monotone (Def. 5), like the hypervolume archiver AA_S. However, unlike the AA_S, MGA does not calculate this unary indicator explicitly, which makes it computationally more tractable than AA_S. In particular, its time complexity is $O(d \cdot N^2 \cdot L)$, where L is the length of the binary encoded input, therefore polynomial.

4 Empirical Study

Despite their crucial importance in the quality of MOEAs, there is surprisingly little experimental work on the behaviour of different archivers [16,13,6]. We provide in this section experiments that confirm the observations in the previous sections, and illustrate some properties of popular archivers that have not been described in the literature.

We have implemented the various archiving algorithms in C++ within a common framework. We make available the initial version of this framework at `http://iridia.ulb.ac.be/~manuel/archivers` in order to help future analysis. We plan to extend this framework in the future with other archivers found in the literature.

In this section, we empirically analyse the reviewed archiving algorithms. In order to focus on the properties of the algorithms, we study the performance of the algorithms when presented with particular sequences of points. The sequences studied have been generated in order to highlight some characteristics of the algorithms.

We evaluate the quality of the algorithms with respect to the hypervolume and unary ϵ-indicator. In all sequences, we run the unbounded algorithm and keep the Pareto front at each iteration of the sequence. In the case of the additive ϵ measure (ϵ_{add}), the reference set is the optimal PF (which is the final unbounded archive). Then, for each archiver and at each iteration, we calculate $\epsilon_{add}(A_t) - \epsilon_{add}(\mathsf{Unbounded}_t)$. Similarly, for the hypervolume we calculate the reference point over the final unbounded Pareto front as

$$r_i = \max f_i + ((1 + (1/(N-1))) \cdot (\max f_i - \min f_i)).$$

Then, we calculate the ratio $\mathrm{HYP}(A_t)/\mathrm{HYP}(\mathsf{Unbounded}_t)$, for each iteration t of the input sequence.

In all sequences, the objective functions are to be minimized, without loss of generality since the sequences are finite, we could always transform them into an all-positive maximization problem and the results will stand.

4.1 MGA addresses key weakness of ϵ-archivers

In both ϵ-approximate and ϵ-Pareto algorithms, the ϵ may become arbitrarily large with respect to the extent of the Pareto front. Knowles and Corne [14] showed that this occurs, for example, when the initial range of objective values is much larger than the actual range of the Pareto front. In that case, the initial estimate of ϵ is much larger than actually needed, but since ϵ cannot decrease, the algorithms end up accepting fewer points than N. This situation occurs even with a small initial estimate of $\epsilon = 0.0001$, as we use in the experiments here. We ran experiments on two sequences proposed by Knowles and Corne [14], of length 10 000 and dimensions 2 and 3, respectively. Fig. 2 and Fig. 3 show that these sequences are not a problem for MGA. Moreover, while MGA is able to maintain an archive size of $|A| = 10$, ϵ-approximate and ϵ-Pareto only keep 2 and 1 solutions respectively just after 4 000 iterations until the end.

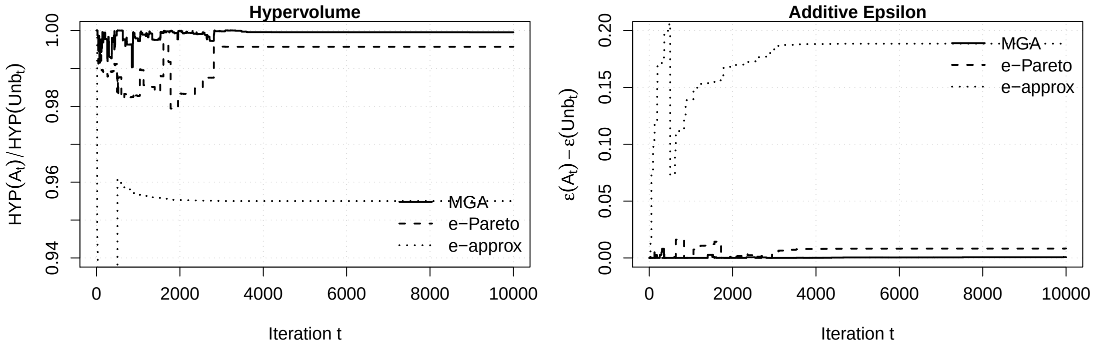


Fig. 2. Small PF (2D) $N = 10$

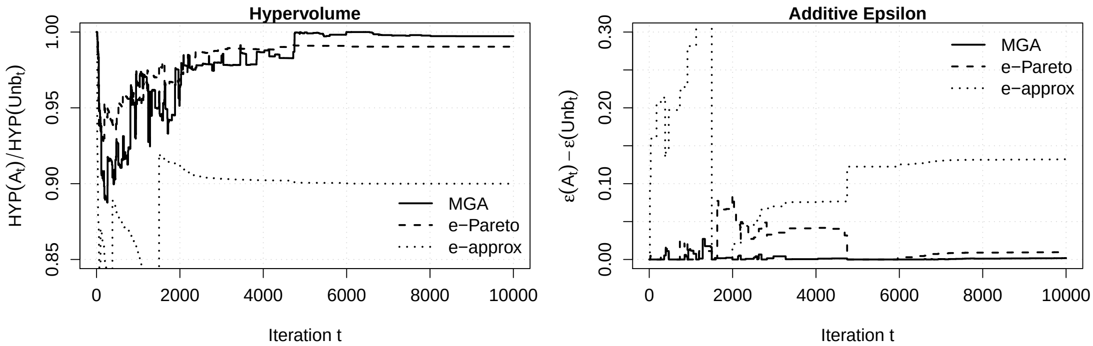


Fig. 3. Small PF (3D) $N = 10$

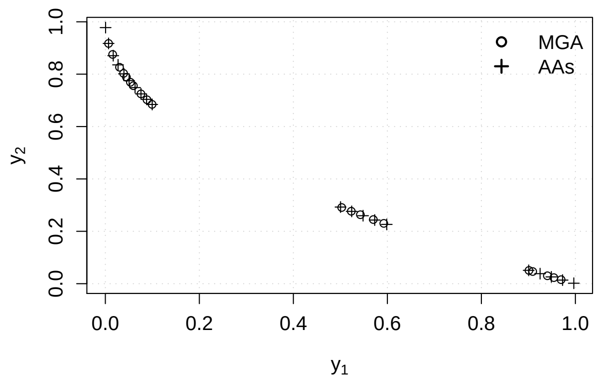
4.2 MGA vs. $\mathbf{AA}_s$ for clustered points

We use a clustered sequence of 900 points in two dimensions to show the different final sets archived by AA_s and MGA. Fig. 4 shows that AA_s keeps the extremes of each cluster, whereas MGA points are not sparsely distributed within each cluster. The result is that AA_s obtains better value in all performance indicators.

4.3 Fast degradation of the SPEA2 archiver

We illustrate how the quality of SPEA2 archiver can degrade very fast if points are added in the extreme of the Pareto front. We generate a sequence of 2 000 nondominated points in a straight line, sorted in increasing order of their first dimension. The top plots in Fig. 5 show that the quality of the archive stored by SPEA2 archiver degrades very rapidly as the sequence progresses. What is happening is that SPEA2 keeps the $N - 1$ initial solutions plus the new extreme, which replaces the old extreme. Therefore, at every step, the gap between the new extreme and the $N - 1$ initial solutions increases further. The final archives are shown in the bottom plot of Fig. 5. All but one solutions archived by SPEA2 are clustered in the left-most extreme of the PF.

The plot also shows that neither MGA nor AA_s obtain a perfect approximation, which for this particular sequence would mean a uniformly distributed archive. Since they do not have knowledge about the real range of the PF, they cannot accurately decide when to keep a solution close to the moving extreme. Nonetheless, MGA and AA_s do not suffer the fast degradation in approximation quality shown by the SPEA2 archiver.



	HYP	ϵ_{add}
AA$_s$	0.9993076	0.009689
MGA	0.9929917	0.013613

Fig. 4. MGA vs. AA$_s$ for clustered points, $N = 20$

4.4 The NSGA-II archiver $\triangleleft$-deteriorates

It is possible to construct a sequence of points such that the NSGA-II archiver removes points from its archive that are Pareto-optimal and includes points that are dominated in such a way that the archived set may be dominated by a previously archived set, and therefore, we say that the quality of the archive has $\triangleleft$-deteriorated over time (Definition 5). Fig. 6 shows the final archive stored by Unbound, AA$_s$, MGA and NSGA-II. Except for the extremes, the rest of the final archive stored by NSGA-II archiver is dominated by solutions stored in previous archives. In fact, for this sequence, the archive at step $t = 58$ is dominated by the archive at step $t = 56$. It is possible to construct different sequences that show the same behaviour for the SPEA2 archiver.

4.5 MGA may deteriorate

In general, MGA may deteriorate (Def. 4), since the final archive may contain points that are dominated by points that were previously in the archive and deleted. This is exemplified in the sequence shown in Fig. 7 for $N = 4$. In this sequence, MGA deletes point d after archiving point e. Then, a and b become dominated by f, and g is accepted into the archive, despite it is dominated by d. A sequence showing that AA$_s$ also deteriorates can be constructed by placing g such that it is dominated only by c.

5 Conclusions

In this paper we have examined the problem of keeping a bounded size approximation of the Pareto front of a set of points in a d-dimensional (objective) space, when the elements of the set are only accessed one-by-one. This models the *archiving* process of keeping an elite population or bounded size "best-so-far" outcome in many multi-objective optimizers.

Earlier works on this problem have dealt with algorithms designed to be stand-alone archivers, such as AGA and ϵ-based archivers. However, the diversity mechanisms employed by popular MOEAs are also archiving algorithms. In this paper, we have proposed a classification of both kinds of archivers, and the recently proposed MGA, according to a number of properties not considered before for this problem (summarised in Table 1). In particular, we differentiate between negative efficiency preservation, monotonicity and $\triangleleft$-monotonicity, and identify two classes of archivers, one based on compatible indicators (hypervolume-based AA$_s$ and the new MGA), and

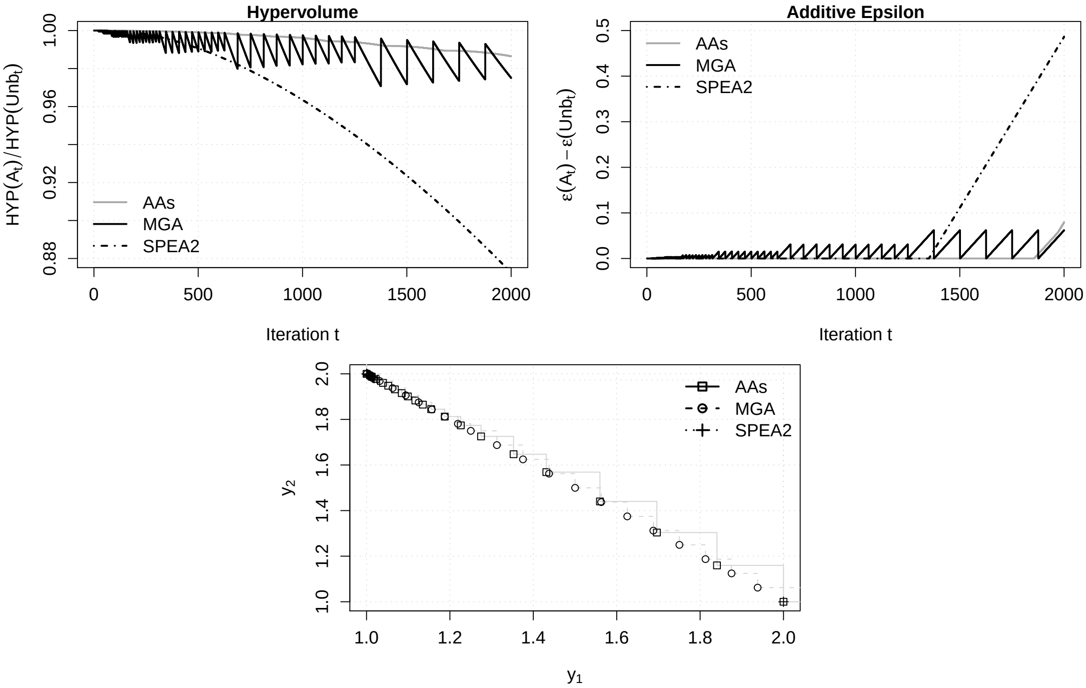


Fig. 5. Increasing extremes (2D) affects SPEA2 performance, $N = 20$

another based on diversity mechanisms (SPEA2, NSGA-II and AGA). This allows us to understand more precisely why the former class of archivers have better convergence properties than the latter class, even when points are seen just once. The former cannot $\lhd$-deteriorate, even if a single point in the archive can be dominated by one that was previously in the archived set. This classification raises the question as to whether there may exist an archiver of the same class as MGA and AA_s that is also monotone.

In addition, our experiments have shown that the recently proposed MGA addresses the key weakness of the earlier ϵ-based archivers, however, at the cost of losing the guarantee of only archiving Pareto-optimal solutions. As a final observation, we did not find an absolute winner, but a tentative assessment is that AA_s often produces better results with respect to hypervolume, whereas MGA often obtains the best ϵ-measure values.

This paper has shown that the archiving problem is far from being well-understood, and we have left open a number of questions. First, we have only examined artificial sequences, designed to show the properties defined here. An interesting extension is to assess the typical performance of the archivers on multiple runs for various simple geometric sequences in varying dimensions, and also from points coming from stochastic search on standard benchmark problems. Second, we have limited ourselves to one-by-one archiving of points and (mostly) a one-pass setting. We know that updating the archive with more than one point simultaneously cannot be a worse approach, and for hypervolume it has already been shown to be superior. Therefore, understanding how the properties defined here extend to other update scenarios is an open research question. Third, we plan to extend this work to other archivers found in the literature, and to foster that project we also provide the archivers and artificial sequences used here to the community.[5] Fourth, we

[5] Source code is available from `http://iridia.ulb.ac.be/~manuel/archivers` and sequences from `http://iridia.ulb.ac.be/~manuel/archivers-sequences`

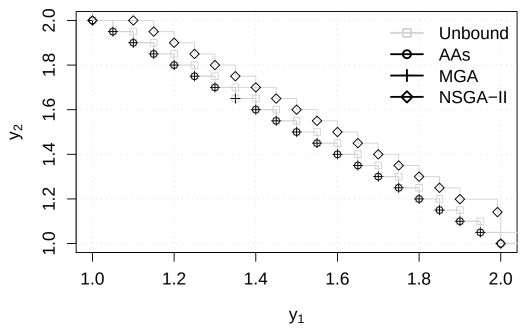


Fig. 6. NSGA-II may $\lhd$-deteriorate over time, $N = 20$

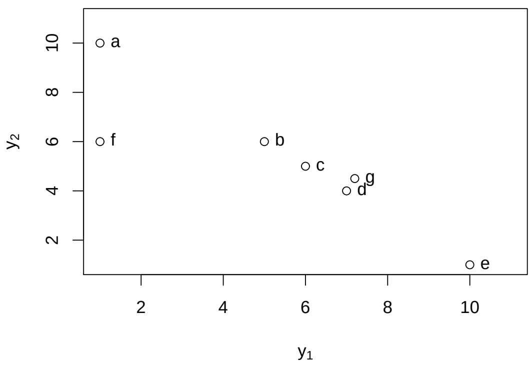


Fig. 7. The final archive of MGA with $N = 4$ is $\{c, e, f, g\}$, which shows that it deteriorates.

plan to use competitive analysis techniques from the field of online algorithms to obtain worst-case bounds, in terms of a measure of "regret" for archivers. Finally, after highlighting some weaknesses of existing archivers, we ask whether designing a better archiver is possible, and what trade-offs exist in its design.

Acknowledgments. Manuel López-Ibáñez acknowledges support from the FRFC project "*Méthodes de recherche hybrides pour la résolution de problèmes complexes*".

References

1. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research 181(3), 1653–1669 (2007)
2. Borodin, A., El-Yaniv, R.: Online computation and competitive analysis. Cambridge University Press, New York, NY (1998)
3. Bringmann, K., Friedrich, T.: Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. In: Ehrgott, M., Fonseca, C.M., Gandibleux, X., Hao, J.K., Sevaux, M. (eds.) Evolutionary Multi-criterion Optimization (EMO 2009), Lecture Notes in Computer Science, vol. 5467, pp. 6–20. Springer, Heidelberg, Germany (2009)
4. Bringmann, K., Friedrich, T.: Don't be greedy when calculating hypervolume contributions. In: Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms (FOGA). pp. 103–112 (2009)
5. Bringmann, K., Friedrich, T.: The maximum hypervolume set yields near-optimal approximation. In: Pelikan, M., Branke, J. (eds.) GECCO 2010, pp. 511–518. ACM press, New York, NY (2010)
6. Corne, D., Knowles, J.D.: Some multiobjective optimizers are better than others. In: Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003), vol. 4, pp. 2506–2512. IEEE Press, Piscataway, NJ (December 2003)
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 181–197 (2002)
8. Fleischer, M.: The measure of Pareto optima. applications to multi-objective metaheuristics. In: Fonseca, C.M., et al. (eds.) Evolutionary Multi-criterion Optimization (EMO 2003), Lecture Notes in Computer Science, vol. 2632, pp. 519–533. Springer, Heidelberg, Germany (2003)
9. Hanne, T.: On the convergence of multiobjective evolutionary algorithms. European Journal of Operational Research 117(3), 553–564 (1999)

10. Hansen, M.P.: Metaheuristics for multiple objective combinatorial optimization. Ph.D. thesis, Institute of Mathematical Modelling, Technical University of Denmark (March 1998)
11. Knowles, J.D.: Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization. Ph.D. thesis, University of Reading, UK (2002)
12. Knowles, J.D., Corne, D.: On metrics for comparing non-dominated sets. In: Proceedings of the 2002 Congress on Evolutionary Computation Conference (CEC'02), pp. 711–716. IEEE Press, Piscataway, NJ (2002)
13. Knowles, J.D., Corne, D.: Properties of an adaptive archiving algorithm for storing nondominated vectors. IEEE Transactions on Evolutionary Computation 7(2), 100–116 (April 2003)
14. Knowles, J.D., Corne, D.: Bounded Pareto archiving: Theory and practice. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) Metaheuristics for Multiobjective Optimisation, pp. 39–64. Lecture Notes in Economics and Mathematical Systems, Springer Verlag, Berlin, Germany (2004)
15. Laumanns, M.: Stochastic convergence of random search to fixed size Pareto set approximations. Arxiv preprint arXiv:0711.2949 (2007)
16. Laumanns, M., Thiele, L., Deb, K., Zitzler, E.: Combining convergence and diversity in evolutionary multiobjective optimization. Evolutionary Computation 10(3), 263–282 (2002)
17. Rudolph, G., Agapie, A.: Convergence properties of some multi-objective evolutionary algorithms. In: Proceedings of the 2000 Congress on Evolutionary Computation (CEC'00). vol. 2, pp. 1010–1016. IEEE Press, Piscataway, NJ (July 2000)
18. Veldhuizen, D.A.V., Lamont, G.B.: Evolutionary computation and convergence to a Pareto front. In: Koza, J.R. (ed.) Late Breaking Papers at the Genetic Programming 1998 Conference. pp. 221–228. Stanford University Bookstore, Stanford University, California (July 1998)
19. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Giannakoglou, K., et al. (eds.) Evolutionary Methods for Design, Optimisation and Control. pp. 95–100. CIMNE, Barcelona, Spain (2002)
20. Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms - A comparative case study. In: Eiben, A.E., et al. (eds.) Parallel Problem Solving from Nature, PPSN V. Lecture Notes in Computer Science, vol. 1498, pp. 292–301. Springer, Heidelberg, Germany (1998)
21. Zitzler, E., Thiele, L., Bader, J.: SPAM: Set preference algorithm for multiobjective optimization. In: Rudolph, G., et al. (eds.) Parallel Problem Solving from Nature, PPSN X, Lecture Notes in Computer Science, vol. 5199, pp. 847–858. Springer, Heidelberg, Germany (2008)
22. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Grunert da Fonseca, V.: Performance assessment of multiobjective optimizers: an analysis and review. IEEE Transactions on Evolutionary Computation 7(2), 117–132 (2003)

Errata Revision 002 (March 2011)

- Section 3.1 defined "negative efficiency preserving" in a way inconsistent with Hanne's original definition. In fact, negative efficiency preserving is equivalent to efficiency preserving when full. A new property "monotone" corresponds now with our previous (wrong) definition of negative efficiency preserving.
- Section 4.1 claimed that MGA maintains an archive size of 20 points. This is obviously wrong since the maximum archive size in those experiments is 10, which is also the correct value of MGA archive size.