technische universität
dortmund

On Multiobjective Selection for Multimodal Optimization

Simon Wessing
Mike Preuss

Algorithm Engineering Report
TR14-2-001
Dec. 2014
ISSN 1864-4503

Faculty of Computer Science
Algorithm Engineering (Ls11)
44221 Dortmund / Germany
http://ls11-www.cs.uni-dortmund.de/

On Multiobjective Selection for Multimodal Optimization

Simon Wessing Mike Preuss

Multiobjective selection operators are a popular and straightforward tool for preserving diversity in evolutionary optimization algorithms. One application area where diversity is essential is multimodal optimization with its goal of finding a diverse set of either globally or locally optimal solutions of a single-objective problem. We therefore investigate multiobjective selection methods that identify good quality and diverse solutions from a larger set of candidates. Simultaneously, unary quality indicators from multiobjective optimization also turn out to be useful for multimodal optimization. We focus on experimentally detecting the best selection operators and indicators in two different contexts, namely a one-time subset selection and an iterative application in optimization. Experimental results indicate that certain design decisions generally have advantageous tendencies regarding run time and quality. One such positive example is using a concept of nearest better neighbors instead of the common nearest-neighbor distances.

1 Introduction

The discipline of multiple-criteria decision making (MCDM) generally deals with the identification of a set of good alternatives under consideration of conflicting objectives [26]. Often, there exists a discrete, predefined set of candidates to choose from. A concrete example of such multiobjective selection problems in the real-world is the identification of a good and diverse subset of molecules in drug discovery [25]. If, however, the set of candidates is not explicitly known beforehand, we rather speak of multiobjective optimization. In this case, the number of alternatives may be either very large or in the case of real-valued decision variables even infinite. Metaheuristics such as evolutionary algorithms (EA) usually attack the multiobjective optimization problem by converting it into a sequence of discrete decision-making problems, where the alternatives to the selection in each iteration are generated by randomly altering the selected solutions of the previous iteration. Multiobjective approaches have even proved beneficial for single-objective optimization problems when a compromise between exploitation and exploration is sought, e. g., in global and multimodal optimization (MMO) [13, 28, 39], which is the focus of our work.

The general approach of applying multiobjective methods to single-objective problems is called multiobjectivization, although the method proposed here does not belong into the originally defined categories of decomposing a scalar function [18] or adding (static) objectives [5]. Earlier attempts of using multiobjective methods for multimodal optimization (as, e. g., [31]) usually belong to the latter category and would exploit a certain problem knowledge. Instead, we are considering a subset selection algorithm incorporating multiobjectivization. Such an approach may be used for arbitrary decision making problems or for survivor selection in evolutionary algorithms. The secondary objective is based on distances to other solutions (neighbors). Therefore, the approach is population-dependent and thus dynamic [48], but problem-independent. A niching effect is achieved without an explicit notion of local optima, and thus the method does not guarantee that all local optima are preserved during selection. Nevertheless, we will show that it is a very good and simple solution if one does not desperately try to locate the possibly huge, complete set of optima a problem possesses but a large subset of very good ones. The approach is versatile and elegant, because it works on non-differentiable problems, is parameterless, and requires neither additional function evaluations nor modifications of the dominance relation. It is therefore well suited to be utilized inside more complex multimodal optimization algorithms. However, here we rather focus on the fundamental differences between multiobjective selection mechanisms and single-objective ones.

Multimodal optimization differs from global optimization in the expected result. The latter only searches for one single globally optimal solution, while the former aims to find a set of (potentially only locally) optimal solutions. This definition is intentionally kept abstract, as any further characterization of this set depends on the user's preferences, which would be represented by using an appropriate *quality indicator* for performance assessment (Sec. 2). We would like to emphasize that we do *not* suggest a concrete, new algorithm for multimodal optimization but rather investigate the effects of different selection variants from a combinatorial space of up to six factors (Sec. 3). We see this as a necessary step towards improving existing and deriving new, better optimization methods. All investigated variants have in common that objective values *and* search space distances are taken into account simultaneously. Some of them are in use already, some are not, and we will use experimental evidence to show which ones are actually useful and which should rather be disregarded.

Parts of this work draw on material from [33], where some of the performance measures were introduced, and [53], where a subset of the selection variants discussed here were defined. However, this is the first time that both topics are connected in two large, computationally involving experimental investigations. These target the interplay of multiobjectivization-based selection mechanisms with different performance measures in the context of i) a simple subset selection problem ("what part of the possibly large result set do we present to the user?") in Sec. 4.2 and ii) an optimization scenario using evolutionary algorithms in Sec. 4.3. For the first time, additional implementation details as the number of neighbors, the use of archives, and greedy behavior are considered. We also rate all these variants regarding elitism (i. e., the ability to keep the best solution in the population). Overall, we strive for obtaining general guidelines concerning selection mechanisms and performance measures for the two mentioned scenarios.

1.1 Niching, Additional Objectives and Novelty Search

Successfully employing multiobjective techniques for multimodal optimization requires bringing together several strands of former and recent developments that were mostly kept separate before. This applies to niching concepts as well as to algorithms utilizing additional objectives in the context of dynamic optimization, and also to the more recently proposed novelty search.

The term *niching* is commonly used to describe diversity maintenance in evolutionary algorithms. Employing the distance to the nearest neighbor in a population of search points $\mathcal{P}$ within selection is a simple possibility to increase or maintain diversity. This has been implemented already by De Jong's famous *crowding* method [11] in a very direct form (comparison with the nearest individual of a fraction of the parent population). Mahfoud [24] later on suggested to consider only the direct parents and compare with the nearest one. Subpopulation (speciation) oriented evolutionary algorithms may rather be seen as following the path of Richardson and Goldberg's *fitness sharing* who define a distance between search points as far or near, according to some distance parameter. While the measures to achieve the separation and the separate development of the subpopulations are different, nearly all attempts either need a given niche distance [29, 22, 41] or use additional evaluations in order to detect which subpopulation a search point belongs to as [51, 44]. In the former case, one is not interested in the nearest neighbor within the population but rather the nearest neighbor in the much smaller set of niche centers. A large survey about conventional approaches for multimodal optimization can be found in [9].

In the recent years, there was a surge of publications using multiobjective selection for solving single-objective problems in general [39], and especially as a means of preserving (decision-space) diversity in the population. To resolve the conflict between the objectives, most of the approaches in the following survey use non-dominated sorting as the first and crowding distance as the secondary selection criterion. Before the concept was employed in single-objective optimization, de Jong et al. [10] applied it to genetic programming and Toffolo and Benini [46] in multiobjective optimization. Bui et al. [6] established the first larger comparison of different secondary objectives. They compared the distance to the nearest neighbor, the average distance to all neighbors, the distance to the best individual, and other objectives not based on distance in the context of dynamic optimization. According to their result, especially the first two performed well. Segredo et al. [38] and Segura et al. [40] continued the research into these variants on static problems with a high number of decision variables and large budgets of function evaluations. In the latter paper, they also investigated the influence of incremental (i. e., less greedy) fitness calculations. Tran et al. [48] compared an NSGA2, using the distance to the nearest neighbor as a second objective, to state-of-the-art algorithms for single-objective optimization. Their experimental setup was according to the black-box optimization benchmarking (BBOB) rules, and therefore aiming at global optimization. However, all these approaches focusing on a single global optimum suffer from a lack of systematic analysis of the diversity preservation capabilities of the algorithms.

Deb and Saha [13] considered multimodal optimization with the objective of discovering all local optima. Their basic idea was to exploit gradient information by considering it as a second objective function. A slightly modified dominance relation ensured that all optima were non-dominated. To get rid of the dependency on derivatives, they also proposed a variant with an integrated local search, consuming $2n$ additional function evaluations for each individual (n standing for the number of search space dimensions). The secondary criterion in this case was the number of better neighbors of a solution. This local search allowed approximating optima to a high precision, but also incurred a high cost. It also introduced several parameters that were later more or less eliminated by means of heuristics [36]. Finally, the approach was abandoned in favor of a simpler selection that considers a solution's average distance to all other population members [1]. Basak et al. [2] use the same second objective, but employ the hypervolume contribution instead of crowding distance to sort non-dominated fronts.

In parallel to these developments, an almost identical approach evolved from the novelty search concept of Lehman and Stanley [20]. The idea of novelty search is to completely disregard the objective function in favor of a solution's novelty, which is defined as the average distance to k nearest neighbors in some phenotypic or behavioral space. As the reference to these spaces already indicates, novelty search was designed to evolve complex structures such as robot controllers. Mouret [28] used multiobjectivization to combine novelty search with common objective-based optimization, obtaining better results than with either of the pure strategies. These results were largely confirmed by [21]. In the novelty search community, it is also customary to use an archive of previous solutions to avoid rediscovering the same areas over and over, a concept that reminds of tabu search [17] and related approaches as, e. g., sequential niching [3]. Although there exists no sensible distinction between genotype, phenotype, or behavior in our case of numerical optimization, we will nonetheless adopt and test the remaining ideas, as has already been done for archiving in [2].

Ulrich and Thiele [50] use an individual's contribution to a diversity indicator value as an objective. This has the drawback that selection has at least cubic run time in the number of individuals. Instead of doing non-dominated sorting, their algorithm alternates between optimizing the original objective and diversity, basically carrying out an iterated level set approximation. As a goal, they correspondingly aim to obtain a maximally diverse set of solutions with objective values below some threshold. The approach is tested on binary and on structural optimization problems.

1.2 Considering the Nearest Better Neighbors

Instead of considering all neighbors, one may especially be interested in the distance to the nearest better neighbor (as all search points carry a quality information). These distances are rather larger than the nearest neighbor differences, the more so, the better a search point is. Nearest-better-neighbor information was first employed in the niching context by [32]. They used the values of the distance from a search point $\boldsymbol{x}$ to the nearest better neighbor in a population $\mathcal{P}$ to estimate a problem's number of basins in an approach called nearest-better clustering (NBC). Points with large nearest-better distance represent the best available approximations to the global/local optima as all the

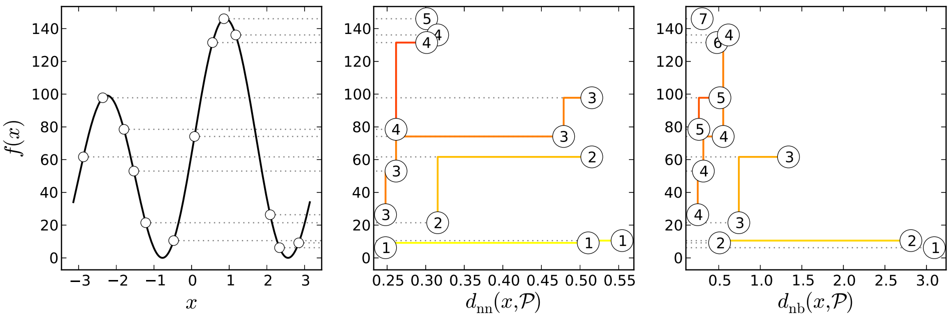

Figure 1: The left panel shows a randomly distributed population on a multimodal landscape, $n = 1$. In the middle, objective values are plotted against each point's distance to the nearest neighbor. The right panel shows the distances to the nearest better neighbor instead. For better visualization, we have set $d_{\mathrm{nb}}(\boldsymbol{x}^*, \mathcal{P})$ to a value of 110% of the largest finite distance value in this figure (it would otherwise be ∞ because there is no better neighbor). Non-dominated fronts are indicated by connecting the respective points and plotting their ranks.

near points must be worse. This idea was combined with multiobjectivization by [53]. The various mentioned distances are generalized to k neighbors and formally defined as follows.

Definition 1. *Let $\boldsymbol{y}_{(1)}, \dots, \boldsymbol{y}_{(\mu)}$ be the ordered elements of $\mathcal{P}$, so that $d(\boldsymbol{x}, \boldsymbol{y}_{(1)}) \leq \dots \leq d(\boldsymbol{x}, \boldsymbol{y}_{(\mu)})$ for some $\boldsymbol{x} \notin \mathcal{P}$. Then, the distance to the k nearest neighbors is denoted $d_{\mathrm{nn}}(\boldsymbol{x}, \mathcal{P}, k) = \frac{1}{k} \sum_{i=1}^{k} d(\boldsymbol{x}, \boldsymbol{y}_{(i)})$.*

Definition 2. *Let $\mathcal{Q} = \{\boldsymbol{y} \in \mathcal{P} \mid f(\boldsymbol{y}) < f(\boldsymbol{x})\}$. As before, assume $\boldsymbol{y}_{(i)} \in \mathcal{Q}$ are ordered so that $d(\boldsymbol{x}, \boldsymbol{y}_{(1)}) \leq \dots \leq d(\boldsymbol{x}, \boldsymbol{y}_{(|\mathcal{Q}|)})$. Then, the distance to the k nearest better neighbors is denoted $d_{\mathrm{nb}}(\boldsymbol{x}, \mathcal{P}, k) = \frac{1}{\min\{k, |\mathcal{Q}|\}} \sum_{i=1}^{\min\{k, |\mathcal{Q}|\}} d(\boldsymbol{x}, \boldsymbol{y}_{(i)})$.*

The special cases of $k = 1$ are consistent with the classic definition of the distance to the nearest neighbor of $\boldsymbol{x}$, $d_{\mathrm{nn}}(\boldsymbol{x}, \mathcal{P}) = \min\{d(\boldsymbol{x}, \boldsymbol{y}) \mid \boldsymbol{y} \in \mathcal{P} \setminus \{\boldsymbol{x}\}\}$, and the distance to the nearest better neighbor $d_{\mathrm{nb}}(\boldsymbol{x}, \mathcal{P}) = \min\{d(\boldsymbol{x}, \boldsymbol{y}) \mid f(\boldsymbol{y}) < f(\boldsymbol{x}) \wedge \boldsymbol{y} \in \mathcal{P}\}$. The nearest- and nearest better neighbor will be denoted with $\mathrm{nn}(\boldsymbol{x}, \mathcal{P})$ and $\mathrm{nbn}(\boldsymbol{x}, \mathcal{P})$, respectively, by using the above definitions with $\arg\min$ instead of $\min$. As we are dealing with minimization problems $f : \mathbb{R}^n \to \mathbb{R}$, we assume the euclidean distance $\|\boldsymbol{x} - \boldsymbol{y}\|_2$ as the underlying distance measure $d(\boldsymbol{x}, \boldsymbol{y})$. In general, however, any measure is possible. The best solutions $\boldsymbol{x}^*$ regarding the objective value have no nearest better neighbor, so we choose $d_{\mathrm{nb}}(\boldsymbol{x}^*, \mathcal{P}, k) := \infty$. This makes them the only non-dominated solutions in multiobjective approaches with $f(\boldsymbol{x})$ and $d_{\mathrm{nb}}(\boldsymbol{x}, \mathcal{P}, k)$ as criteria (but note that they would be part of the non-dominated set anyway due to their extremal objective value).

Figure 1 shows a randomly distributed population spread out on a multimodal landscape. This simple example already indicates that a multiobjective selection procedure based on non-dominated ranking is in principle able to maintain several optima at once. However, we will also have to specify a preference between incomparable solutions within a non-dominated front to arrive at an unambiguous definition. Apparently, this preference mostly has theoretical influence, as results in Sec. 3 and 4 show. But first we will discuss the current state-of-the-art in performance assessment for multimodal optimization in detail.

2 Performance Assessment

We dismiss the idea of formulating a single success criterion (e. g., “finds all optima”) for multimodal optimization algorithms, because, firstly, it seems implausible that every local optimum is equally important, and secondly, the task of finding all optima with high precision may be too challenging. We speak of a *floor effect* when a measurement indicates no progress because the posed task is too difficult. Troubles with such kinds of performance measurement have been demonstrated in [53]. Furthermore, diversity and optimality are conflicting and there seems no natural way to aggregate them. A success criterion is only binary and thus cannot express the inherently multiobjective character of assessing a population’s features. One possible workaround is to change the problem formulation into a level-set approximation problem. Here, the goal is to maximize the diversity of a set of points with objective values below some threshold [8, 50, 15]. However, specifying a sensible threshold parameter is difficult. Therefore, we present a large collection of quality indicators that can represent the multimodal optimization problem in more detail. Many of these indicators require information about the location (and the basins of attraction) of the optima, and are therefore only suited for benchmarking.

We interpret measuring the quality of a solution set for a multimodal problem as consisting of several stages. At first, an optimization algorithm provides a solution set. This may, e. g., be the final population or an archive of recorded good solutions. The cardinality of this set should depend on the tackled problem, but is at current usually determined by the employed algorithm alone. Thus, the set can become extremely large, which is inappropriate, because it may firstly bias the quality assessment and secondly make a human inspection of the set too laborious and costly. So, we explicitly assume an automated subset selection step as part of the evaluation process. We may term the result of the subset selection a *representing set* or *approximation set*, and it is important to note that all the other solutions contained in the solution set have no influence on the result of the measurement.

The term *quality indicator* is taken from multiobjective optimization [54] and simply describes a mapping that assigns each solution set $\mathcal{P}$ a real number. We classify the presented quality indicators according to the amount of information that is necessary for their application. Throughout the section, $\mathcal{P} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_\mu\}$, $\mu < \infty$, denotes the approximation set that is to be assessed. Some of the indicators already exist for some time, while others were recently presented in [33]. So, no comprehensive analysis of the whole collection existed so far.

2.1 Indicators without Problem Knowledge

Solow-Polasky Diversity (SPD) Solow and Polasky [43] developed an indicator to measure a population's biological diversity and showed that it has superior theoretical properties compared to the sum of distances and other indicators. Ulrich et al. [49] discovered its applicability to multiobjective optimization. They also verified the inferiority of the sum of distances experimentally by directly optimizing the indicator values. To compute this indicator for $\mathcal{P}$, it is necessary to build a $\mu \times \mu$ correlation matrix $\mathbf{C}$ with entries $c_{ij} = \exp(-\theta d(\boldsymbol{x}_i, \boldsymbol{x}_j))$. The indicator value is then the scalar resulting from $\mathrm{SPD}(\mathcal{P}) := \boldsymbol{e}^\top \mathbf{C}^{-1} \boldsymbol{e}$, where $\boldsymbol{e} = (1, \dots, 1)^\top$. As the matrix inversion requires time $O(\mu^3)$, the indicator is only applicable to relatively small sets. It also requires a user-defined parameter θ, which depends on the size of the search space. We preliminarily choose $\theta = 1/n$ throughout this paper.

Sum of Distances (SD) As already mentioned, the sum of distances

$$\mathrm{SD}(\mathcal{P}) := \sqrt{\sum_{i=1}^{\mu} \sum_{j=i+1}^{\mu} d(\boldsymbol{x}_i, \boldsymbol{x}_j)}$$

is criticized by [43, 49, 25] as being inappropriate for a diversity measure, because it only rewards the spread, but not the diversity of a population. The figure should therefore not be used. However, if it is used, we suggest to take the square root of the sum, to obtain indicator values of reasonable magnitude.

Sum of Distances to Nearest Neighbor (SDNN) As [49] showed that SD has some severe deficiencies, we also consider the sum of distances to the nearest neighbor

$$\mathrm{SDNN}(\mathcal{P}) := \sum_{i=1}^{\mu} d_{\mathrm{nn}}(\boldsymbol{x}_i, \mathcal{P}) \, .$$

In contrast to SD, SDNN penalizes the clustering of solutions, because only the nearest neighbor is considered. Emmerich et al. [15] mention the arithmetic mean gap $\frac{1}{\mu}\mathrm{SDNN}(\mathcal{P})$ and two other similar variants. We avoid the averaging here to reward larger sets. However, it is still possible to construct situations where adding a new point to the set decreases the indicator value.

Statistics of the Distribution of Objective Values Regarding the assessment of the population's raw performance, few true alternatives seem to exist. To us, the only things that come to mind are statistics of the objective value distribution, with the mean or median as the most obvious measures. Values from the tail of the distribution, as the best or worst objective value, do not seem robust enough to outliers. Thus we include the average objective value $\mathrm{AOV}(\mathcal{P}) := \frac{1}{\mu} \sum_{i=1}^{\mu} f(\boldsymbol{x}_i)$ as a representative for this category in our experiments.

2.2 Indicators Requiring Knowledge of the Optima

All indicators in this section require a given set of locally optimal positions $\mathcal{Q} = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_m\}$, $m < \infty$, to assess $\mathcal{P}$. This means they can only be employed in a benchmarking scenario on test problems that were specifically designed so that $\mathcal{Q}$ is known. Note, however, that $\mathcal{Q}$ does not necessarily have to contain all existing optima, but can also represent a subset (e. g., only the global ones).

Peak Ratio (PR) Ursem [51] defined the number of found optima $\ell = |\{\boldsymbol{z} \in \mathcal{Q} \mid d_{\text{nn}}(\boldsymbol{z}, \mathcal{P}) \leq \epsilon\}|$ divided by the total number of optima as peak ratio $\text{PR}(\mathcal{P}) := \ell/m$. The indicator requires some constant ϵ to be defined by the user, to decide if an optimum has been approximated appropriately.

Peak Distance (PD) This indicator simply calculates the average distance $\text{PD}(\mathcal{P}) := \frac{1}{m}\sum_{i=1}^{m} d_{\text{nn}}(\boldsymbol{z}_i, \mathcal{P})$ of a member of the reference set $\mathcal{Q}$ to the nearest individual in $\mathcal{P}$. A first version of this indicator (without the averaging) was presented by [44] as "distance accuracy". With the $1/m$ part, peak distance is analogous to the indicator inverted generational distance [7], which is computed in the objective space of multiobjective problems.

Peak Inaccuracy (PI) Thomsen [45] proposed the basic variant of the indicator

$$\text{PI}(\mathcal{P}) := \frac{1}{m}\sum_{i=1}^{m} |f(\boldsymbol{z}_i) - f(\text{nn}(\boldsymbol{z}_i, \mathcal{P}))|$$

under the name "peak accuracy". To be consistent with PR and PD, we also add the $1/m$ term here. We allow ourselves to relabel it to peak inaccuracy, because speaking of accuracy is a bit misleading as the indicator must be minimized. PI has the disadvantage that the representativeness of $\mathcal{P}$ is not directly rewarded, because it is possible for one solution to satisfy several optima at once. Note that PI is somewhat related to the maximum peak ratio (MPR) by [27]. MPR is also extensively used by [42].

Averaged Hausdorff Distance (AHD) This indicator can be seen as an extension of peak distance due to its relation to the inverted generational distance. It was defined by [37] as

$$\Delta_p(\mathcal{P}, \mathcal{Q}) = \max\left\{\left(\frac{1}{m}\sum_{i=1}^{m} d_{\text{nn}}(\boldsymbol{z}_i, \mathcal{P})^p\right)^{1/p}, \left(\frac{1}{\mu}\sum_{i=1}^{\mu} d_{\text{nn}}(\boldsymbol{x}_i, \mathcal{Q})^p\right)^{1/p}\right\}.$$

The definition contains a parameter p that controls the influence of outliers on the indicator value (the more influence the higher p is). For $1 \leq p < \infty$, AHD has the property of being a semi-metric [37]. We choose $p = 1$ throughout this paper, analogously to [15]. The practical effect of the indicator is that it rewards the approximation of the optima (as PD does), but as well penalizes any unnecessary points in remote locations.

This makes it an adequate indicator for the comparison of approximation sets of different sizes.

2.3 Indicators Requiring Knowledge of the Basins

Even more challenging to implement than the indicators in Sec. 2.2 are indicators that require information about which basin each point of the search space belongs to.

Definition 3 (Attraction basin, [47]). *For the position $\boldsymbol{z} \in \mathbb{R}^n$ of an optimum,* $\mathrm{basin}(\boldsymbol{z}) \subseteq \mathbb{R}^n$ *is the largest set of points such that for any starting point* $\boldsymbol{x} \in \mathrm{basin}(\boldsymbol{z})$ *the infinitely small step steepest descent algorithm will converge to* $\boldsymbol{z}$*.*

In practice, the set $\mathrm{basin}(\boldsymbol{z})$ can only be approximated, either by using a carefully constructed test problem, or by running a hill climber for each $\boldsymbol{x} \in \mathcal{P}$ as a start point during the assessment and then matching the obtained local optima with the points of the known $\mathcal{Q}$. Regardless of how it is achieved, we assume the existence of a function

$$b(\boldsymbol{x}, \boldsymbol{z}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \mathrm{basin}(\boldsymbol{z}), \\ 0 & \text{else.} \end{cases}$$

The rationale of indicators for covered basins instead of distances to local optima is that the former also enables measuring in early phases of an optimization, when the peaks have not been approximated well yet. If the basin shapes are not very regular, the latter indicator type may be misleading in this phase.

Basin Ratio (BR) The number of covered basins is calculated as

$$\ell = \sum_{i=1}^{m} \min\{1, \sum_{j=1}^{\mu} b(\boldsymbol{x}_j, \boldsymbol{z}_i)\} .$$

The basin ratio is then $\mathrm{BR}(\mathcal{P}) := \ell/m$, analogous to PR. This indicator can only assume $m+1$ distinct values, and in lower dimensions it should be quite easy to obtain a perfect score by a simple random sampling of the search space. It makes sense especially when not all of the existing optima are relevant. Then, its use can be justified by the reasoning that the actual optima can be found relatively easily with a hill climber, once there is a start point in each respective basin.

Basin Inaccuracy (BI) This combination of basin ratio and peak inaccuracy was proposed by [33]. It is defined as

$$\mathrm{BI}(\mathcal{P}) := \frac{1}{m} \sum_{i=1}^{m} \begin{cases} \min\left\{|f(\boldsymbol{z}_i) - f(\boldsymbol{x})| \mid \boldsymbol{x} \in \mathcal{P} \wedge b(\boldsymbol{x}, \boldsymbol{z}_i)\right\} & \exists \boldsymbol{x} \in \mathrm{basin}(\boldsymbol{z}_i) , \\ f_{\max} & \text{else,} \end{cases}$$

where $f_{\max}$ denotes the difference between the global optimum and the worst possible objective value. For each optimum, the indicator calculates the minimal difference in

Table 1: Overview of the quality indicators.

Ind.	Best	Worst	Uses $f(\boldsymbol{x})$	Use with var. μ	w/o optima	w/o basins	w/o params
SPD	μ	1	✘	✔	✔	✔	✘
SD	> 0	0	✘	✘	✔	✔	✔
SDNN	> 0	0	✘	✘	✔	✔	✔
AOV	$f(\boldsymbol{x}^*)$	$> f(\boldsymbol{x}^*)$	✔	✔	✔	✔	✔
PR	1	0	✘	✘	✘	✔	✘
PD	0	> 0	✘	✘	✘	✔	✔
PI	0	> 0	✔	✘	✘	✔	✔
AHD	0	> 0	✘	✔	✘	✔	✘
BR	1	0	✘	✘	✘	✘	✔
BI	0	> 0	✔	✘	✘	✘	✔

objective values between the optimum and all solutions that are located in it's basin. If no solution is present in the basin, a penalty value is assumed for it. Finally, all the values are averaged. The rationale behind this indicator is to enforce a good basin coverage, while simultaneously measuring the deviation of objective values $f(\boldsymbol{x})$ from $f(\boldsymbol{z}_i)$.

2.4 Summary

The ideal indicator for multimodal optimization would probably regard both diversity and objective values, enable fair comparisons of sets with different sizes, and require no problem information or additional parameters. Table 1 shows a classification of the indicators regarding these properties. With the notable exception of AOV and AHD, all presented indicators are inclined to favor approximation sets with unduly large sizes μ. While this makes sense for diversity indicators, it should be avoided for those in Sec. 2.2 and 2.3.

Unfortunately, only diversity indicators and AOV can be applied in real-world applications, as they are the only ones not needing any problem knowledge. In benchmarking scenarios, we have more options available, although care has to be taken not to conduct unfair comparisons. AHD is more challenging than PD, because the former not only rewards the approximation of $\mathcal{Q}$, but also penalizes superfluous points in remote locations. AHD's parameter should be well-tempered. While PR has a straightforward interpretation, it can be easily misconfigured [53]. Therefore, BR is a good parameterless alternative, especially if only a subset of all optima is requested to be found, as in [23]. BR and BI also put a higher emphasis on diversity than the "peak-oriented" indicators, but are more difficult to implement, because they rely on more problem knowledge. PI can be easily deceived when an optimum is not covered by any solution, but another similarly good solution exists in another basin nearby [33].

Table 2: Examined selection variants

	High Priority	Neighbor	Multiobjective
SV1	objective	nearest	false
SV2	objective	nearest	true
SV3	objective	nearest better	false
SV4	objective	nearest better	true
SV5	distance	nearest	false
SV6	distance	nearest	true
SV7	distance	nearest better	false
SV8	distance	nearest better	true
CD-NN	crowding distance	nearest	true
CD-NB	crowding distance	nearest better	true

3 Subset Selection

To be able to carry out systematic experiments, we not only need appropriate quality indicators, but also have to define selection variants (SV) in a way that lets us attribute the observed effects to certain design decisions. In [53], the following three factors were first identified to collectively define a selection criterion:

- *High priority* defines if the objective- or the distance value should be preferred as a selection criterion. In the multiobjective case, also crowding distance is an option.
- The type of *neighbor* information (nearest or nearest better) decides which distance function to use.
- *Multiobjectivization* controls if the two criteria objective value and distance are treated in a multiobjective fashion or only in lexicographic order.

Table 2 lists all 2^3 possible selection variants according to these factors plus two variants based on the popular crowding distance (CD). Hypervolume contribution [4] is omitted in our investigations due to its relative similarity to crowding distance. Subsequently, all these variants are combined with different numbers of neighbors, different approaches regarding the removal of more than one solution (incremental/non-incremental), and archives. In total this amounts to 120 different selection approaches, although some are actually duplicates of each other, as we will see later. The pseudocode in Fig. 1 shows how the parameters control the behavior. When applying the respective selection criteria, the algorithm can either follow a greedy approach by removing one individual per iteration ($r = 1$) and recalculating distance values each time, or it can use a "super-greedy" approach by removing all individuals at once ($r = \lambda$). In the context of hypervolume contribution-based selection, Bringmann and Friedrich [4] discourage the (more expensive) greedy behavior in favor of an even more expensive exact calculation of the optimal subset. However, we disregard this option for now, as it is (yet) unclear which population-based indicator value to optimize. Additionally it would probably be

Algorithm 1 Parameterized selection

Input: Population $\mathcal{P} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_{\mu+\lambda}\}$, archive $\mathcal{A}$, number of considered neighbors k, number of removed individuals per iteration r
Output: Surviving individuals

$\mathcal{Q} \leftarrow \mathcal{P} \cup \mathcal{A}$
if neighbor mode is nearest **then**
  $d(\cdot,\cdot,\cdot) \leftarrow d_{\mathrm{nn}}(\cdot,\cdot,\cdot)$
else if neighbor mode is nearest better **then**
  $d(\cdot,\cdot,\cdot) \leftarrow d_{\mathrm{nb}}(\cdot,\cdot,\cdot)$
end if
while length($\mathcal{P}$) $> \mu$ **do**
  for all $\boldsymbol{x} \in \mathcal{P}$ **do**
    calculate $d(\boldsymbol{x}, \mathcal{Q}, k)$
  end for
  if multiobjective is true **then**
    compute non-dominated fronts $\mathcal{F}_1, \dots, \mathcal{F}_s$
    for all $\mathcal{F}_i \in \{\mathcal{F}_1, \dots, \mathcal{F}_s\}$ **do**
      if high priority is objective **then**
        sort $\mathcal{F}_i$ ascending by objective values $f(\boldsymbol{x})$
      else if high priority is distance **then**
        sort $\mathcal{F}_i$ descending by distance values $d(\boldsymbol{x}, \mathcal{Q})$
      else if high priority is crowding distance **then**
        sort $\mathcal{F}_i$ descending by CD, using the algorithm in [19]
      end if
    end for
    $\mathcal{P} \leftarrow$ concatenate $\mathcal{F}_1, \dots, \mathcal{F}_s$
  else if multiobjective is false **then**
    if high priority is objective **then**
      sort $\mathcal{P}$ by $(f(\boldsymbol{x}), -d(\boldsymbol{x}, \mathcal{Q}, k))$ in ascending lex. order
    else if high priority is distance **then**
      sort $\mathcal{P}$ by $(-d(\boldsymbol{x}, \mathcal{Q}, k), f(\boldsymbol{x}))$ in ascending lex. order
    end if
  end if
  remove last r elements of $\mathcal{P}$
end while
return $\mathcal{P}$

too computationally expensive anyway, as there are $\binom{\mu+\lambda}{\mu}$ subsets of size μ. Analogously to the question if distance values in the search space should be recalculated, the same problem applies in objective space. So far, authors usually employ the original, super-greedy crowding distance procedure [12]. We, however, use the improved, greedy version of [19], which yields a much more uniform sampling of the non-dominated front.

The given algorithm is not tailored towards efficiency but simplicity, and covers all selection variants examined in this paper. As usual, the selection gets the current population $\mathcal{P}$ as input and returns the individuals selected for survival. Optionally, an archive $\mathcal{A}$ of older individuals may be incorporated into the neighborhood calculation, to prevent rediscovering previously visited areas of the search space. Regardless of how the archive

Table 3: Intrinsic elitism of selection variants

	$\mathcal{A} = \emptyset$		$\forall \boldsymbol{a} \in \mathcal{A} : f(\boldsymbol{a}) \geq f(\boldsymbol{x}^*)$		$\exists \boldsymbol{a} \in \mathcal{A} : f(\boldsymbol{a}) < f(\boldsymbol{x}^*)$	
	$k = 1$	$k > 1$	$k = 1$	$k > 1$	$k = 1$	$k > 1$
SV1	✔	✔	✔	✔	✔	✔
SV2	✔	✔	✔	✔	✔	✔
SV3	✔	✔	✔	✔	✔	✔
SV4	✔	✔	✔	✔	✔	✔
SV5	✔	✘	✘	✘	✘	✘
SV6	✔	✘	✘	✘	✘	✘
SV7	✔	✔	✔	✔	✘	✘
SV8	✔	✔	✔	✔	✘	✘

is actually managed in practice, the important characterization for us is that its solutions influence the fitness of members of the population, but do not underlie survivor selection themselves. The archive hopefully enables us to work with smaller population sizes, reducing the cost of the quadratic distance matrix computation. Based on this situation, we make the following observation:

Proposition 1. *If $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{P} : f(\boldsymbol{x}) \neq f(\boldsymbol{y})$, then SV1 and SV3 behave identical. If additionally $k = 1$, $\mathcal{A} = \emptyset$, and incremental selection is used ($r = 1$), then also SV5 and SV7 behave identical.*

Proof. The statement is obvious for SV1 and SV3, because the order is completely given by the objective values. For SV5 and SV7, without loss of generality, let $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{P}$ be the pair of solutions for which $d(\boldsymbol{x}, \boldsymbol{y})$ is minimal and let $f(\boldsymbol{x}) < f(\boldsymbol{y})$. Then, $\boldsymbol{y}$ is in both cases considered the worst solution in the population. This is sufficient to establish the identical behavior, because only one individual at a time is removed. If $\mathcal{A} \neq \emptyset$, we can construct a counterexample for SV5 and SV7 by choosing $\mathcal{A} = \{\boldsymbol{a}\}$ and $\mathcal{P} = \{\boldsymbol{x}, \boldsymbol{y}\}$ for which $f(\boldsymbol{x}) < f(\boldsymbol{a}) < f(\boldsymbol{y})$ and $d(\boldsymbol{x}, \boldsymbol{a}) < d(\boldsymbol{y}, \boldsymbol{a})$. Then SV5 would remove $\boldsymbol{x}$ while SV7 would remove $\boldsymbol{y}$. □

This result is interesting, because it shows that using nearest-better distances alone does not always mean a change, leave alone an improvement. In some cases, a multi-objective approach is necessary to obtain a benefit from nearest-better information in selection. Furthermore, the archive influences all "distance-focused" selection variants (SV5–SV8), regarding their ability to retain all best solutions $\boldsymbol{x}^*$ in the population.

Proposition 2. *SV5 to SV8 do not guarantee to retain all best solutions $\boldsymbol{x}^*$ in a population in the presence of a non-empty archive.*

Proof. In a slightly modified example with $\mathcal{A} = \{\boldsymbol{a}\}$ and $\mathcal{P} = \{\boldsymbol{x}, \boldsymbol{y}\}$, for which $f(\boldsymbol{a}) < f(\boldsymbol{x}) < f(\boldsymbol{y})$ and $d(\boldsymbol{x}, \boldsymbol{a}) < d(\boldsymbol{y}, \boldsymbol{a})$, all selection variants that put a high priority on distance remove $\boldsymbol{x}$, because it has the smaller distance to its nearest (better) neighbor. □

The last two columns in Tab. 3 summarize the statement of Proposition 2. Neither SV5 nor SV7 possess elitism when the archive is generally non-empty. However, there are cases where only SV5 fails, as seen in Proposition 1. This leads us to a further differentiation: The situation that $\exists \boldsymbol{a} \in \mathcal{A} : f(\boldsymbol{a}) < f(\boldsymbol{x}^*)$ can never appear if we start with an empty archive, use a selection that rejects the worse solution in the example in Proposition 1, and subsequently fill the archive with rejected solutions. Thus, in the special situation where it is guaranteed that $\forall \boldsymbol{a} \in \mathcal{A} : f(\boldsymbol{a}) \geq f(\boldsymbol{x}^*)$, also SV7 and SV8 again guarantee to retain the best solution in the population. The elitism of CD-NN and CD-NB stems from their forced selection of the boundary points of the non-dominated front when $\mu + \lambda \geq 3$ (this also holds for appropriate variants based on hypervolume contribution). Generally, it is assumed that elitism is a desired property. Without experiments, however, it is difficult to estimate how the tradeoff between distance and objective value should be chosen, as analogous decisions also have to be made between worse individuals. Please note that retaining the best solutions does not necessarily mean that all locally optimal solutions are kept, once they are in the population. The fitness of each locally optimal individual depends on the depth and size of its basin in relation to the competing ones. Thus, the decision to keep or delete any solution is always dependent on the problem and the current population. Therefore, we will analyze all selection variants experimentally in the following, in order to obtain a better understanding of their behavior.

4 Experiments

There are numerous algorithms for multimodal optimization out there, and with them numerous ways to control the search process, do variation, local search, and archiving. However, all algorithms sooner or later have to cope with the problem of what to keep and what to throw away, namely selection. This problem naturally comes in two flavors: online and offline. The offline situation resembles the subset selection problem, when the optimization is finished and we have to choose a subset of the obtained final population and/or archive. The online situation occurs during the run of the optimization algorithm. We therefore undertake two experiments in order to find out which selection variants are most successful for each of these two scenarios and can therefore be recommended for further use in more complex algorithms for multimodal optimization.

4.1 Test Instances

For our experiments, we need to select some reasonable test problems. However, those in the CEC 2013 benchmark suite [23] do not supply enough information to (efficiently) apply all indicators. Only the positions for a few global optima are known and the corresponding attraction basins would have to be estimated by hill climbing. Furthermore, many instances are only trivial one-dimensional problems. Eiben and Jelasity [14] suggest to use parametrized problem *generators* that can produce test instances randomly, to exercise fine control over problem difficulty. Such generators were also explicitly proposed for multimodal optimization by Rönkkönen et al. [35]. Our choice is the test problem generator "N-Peaks" by [30], which is very similar to the "quadratic family" in [35].

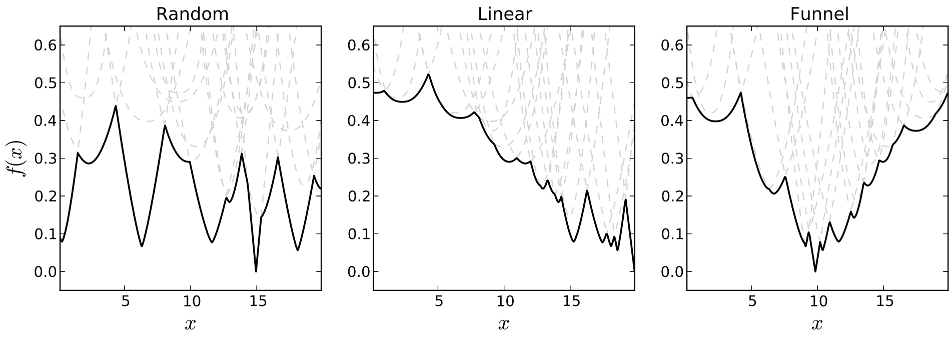

Figure 2: Illustrations of the used test problems. The landscapes are generated by taking the minimum of overlapping unimodal functions.

It produces multimodal problem instances by combining several randomly distributed peaks. The problems are non-separable and irregular, which are also important features of difficult real-world problems. The N-Peaks function itself is defined by the following formulas:

$$f(\boldsymbol{x}) = \min_{\forall \boldsymbol{p}} \{g(\boldsymbol{x}, \boldsymbol{p})\} \tag{1}$$

$$g(\boldsymbol{x}, \boldsymbol{p}) = h_{\boldsymbol{p}} \left(\left(\frac{\mathrm{md}(\boldsymbol{x}, \boldsymbol{p})}{r_{\boldsymbol{p}}} \right)^{s_{\boldsymbol{p}}} - 1 \right) + 1 \tag{2}$$

$$\mathrm{md}(\boldsymbol{x}, \boldsymbol{p}) = \sqrt{\max \left\{ 0, \sum_{i=1}^{n} (x_i - p_i)^2 + \mathrm{dep}(\boldsymbol{x}, \boldsymbol{p}) \right\}} \tag{3}$$

$$\mathrm{dep}(\boldsymbol{x}, \boldsymbol{p}) = \sum_{j=1}^{n} \sum_{k=j+1}^{n} (x_j - p_j)(x_k - p_k) D_{\boldsymbol{p},jk} \tag{4}$$

$$D_{\boldsymbol{p},jk} = u/(n - 1 - j) \tag{5}$$

The objective function is given in (1). It takes the minimum of N unimodal functions (2) around peaks $\boldsymbol{p}$. This has the advantage that local optima with known positions are created, which is in turn necessary to calculate some quality indicators (see Sec. 2.2). Each of these functions (2) is controlled by its own parameters $h_{\boldsymbol{p}}$, $s_{\boldsymbol{p}}$, and $r_{\boldsymbol{p}}$ for depth, shape, and radius, respectively. An example for the instantiation of these parameters can be found in Sec. 4.2. A basin is simply formed by calculating a euclidean distance (3) from $\boldsymbol{p}$, modified by a dependency term (4). Equation 4 in turn requires values $D_{\boldsymbol{p},jk}$, which are calculated in (5) using random numbers $u \sim \mathcal{U}(-0.5, 0.5)$. These are drawn once during initialization and then stored.

At this point we extend the original definition of the generator by adding some rules for the creation of different global structures. For a "linear" topology, the depth values $h_{\boldsymbol{p}}$ are reordered so that we have the i-th largest depth assigned to the peak with the i-th largest value of $\|\boldsymbol{p}\|_1 = \sum_{j=1}^{n} p_j$ the global optimum thereby sitting near to a corner of the domain. For a "funnel" structure, depths are chosen to grow with increasing

Table 4: Additional factors for experiment 4.2

Factor	Type	Symbol	Levels
Number solutions	environmental		$\{125, 250, 500, 1000\}$
Topology	environmental		{random, linear, funnel}
Number variables	environmental	n	$\{2, 3, 5, 10, 20\}$
Number neighbors	control	k	$\{1, 15, \text{all}\}$
Incremental	control		{yes, no}

euclidean distance from $(10, \dots, 10)^\top$ (the centroid of the valid domain). If no depth reordering is carried out, we have the "random" topology. Additionally, shape values $s_{\boldsymbol{p}}$ are always reordered so that they decrease with increasing $h_{\boldsymbol{p}}$. The effect is that the robustness of optima is correlated to their objective value (the global optimum has the least robustness). This is one possible justification for doing multimodal optimization. Figure 2 shows examples for the resulting problems for $n = 1$ and $N = 20$. Note that $m \leq N$, because peaks can be masked by others.

4.2 Subset Selection Experiment

Research Question Which selection variant is the most successful in the situation described as "subset selection" in Sec. 3?

Pre-experimental Planning Corresponding to the chosen scenario, we assume that the number of points in the solution set cannot be controlled. Therefore we test several different set sizes. However, as the run time of most selection variants and indicators is at least quadratic, we cannot consider exceedingly large solution sets. Regarding the necessary problem knowledge, we assume that the number of optima is known and is equal to the number of desired optima. Thus, we will select exactly m solutions from the original set. Note that making different assumptions here should lead to the consideration of completely different selection algorithms. If, for example, the number of desired optima is small but not exactly specified, *representative 5 selection* would be an alternative [33].

Task The obtained representing sets are evaluated with all indicators mentioned in Sec. 2. We regard two configurations as equally effective regarding some indicator, if a Mann-Whitney U test cannot reject the null hypothesis that there is no significant difference between the two with 95% confidence. Of course it is the task of the selection variants to obtain an as good as possible performance regarding all indicators at once. As this is infeasible, we will try to explain the observed effects.

Setup The setting in this experiment is a hypothetical subset selection task at the moment where optimization is finished and we are faced with the problem of identifying interesting solutions among the ones the algorithm generated. To imitate this situation, we first generate a random problem instance. A solution set is constructed for this problem, containing all local optima (between 50 and 100 points) and a larger number of non-optimal solutions that are drawn randomly. The latter represent points that were visited during the algorithm's search for the optima. The set size is then reduced to the size of the representing set either sequentially by removing one individual at a time or by identifying all the worst individuals at once. Apart from the basic selection variants listed in Tab. 2, we consider the factors in Tab. 4 for this experiment. Each factor level is combined with all other possible levels, leading to a full factorial experimental design. Every configuration is replicated eleven times with a random test instance and random solution set. The sample sizes in the statistical tests are $11 \cdot 4 \cdot 3 = 132$, because we also have four solution set sizes and three problem topologies from the environmental factors. The randomness in the test instances results from the following procedure: $N = 100$ peaks are drawn with uniform random positions within $[0, 20]^n$. For each peak $\boldsymbol{p}$, a depth $h_{\boldsymbol{p}} \in [0.5, 0.99]$, a shape $s_{\boldsymbol{p}} \in [1, 3]$, and a radius $r_{\boldsymbol{p}} \in [5\sqrt{n}, 10\sqrt{n}]$ are drawn also uniformly random, followed by the (potential) reordering described in Sec. 4.1. The global optimum has its depth explicitly set to 1. The resulting problems exhibit structures as in Fig. 2, only in higher dimensions.

Results Figure 3 shows performance details of some example configurations. Here, the progress of incremental selection variants with $k = 1$, while reducing the number of solutions from 250, is plotted. The problems belong to the random topology and $n = 5$. To obtain smoother curves, we increased the number of repeats to 100 for these figures. Relative performance is used to better visualize the differences. In each panel, the indicator value of SV1 is taken as a reference, because it is independent of k and incremental selection. In Fig. 4, a level plot of all configurations' final performance is shown. The results are averaged over all solution set sizes, problem topologies, and incremental/non-incremental versions. This is done because the former two are environmental factors and the effects of the latter are difficult to visualize in this form. Table 5 lists the three best configurations (differentiating between incremental/non-incremental) for the indicators AHD, PR, BR, and BI according to best mean performance. None of the differences between the first place and the two lower ranked configurations is statistically significant. (In some cases there were no significant differences between the top eight configurations of the sixty tested.)

Observations Figure 3 shows that non-incremental selection variants tend more to extremes than the incremental ones. An additional analysis of variance shows that the factor interacts with the used distance function and n: When using nearest-better distances, non-incremental selection should be used to obtain good PR and AHD values. The conventional nearest-neighbor distance, on the other hand, should be coupled with incremental selection. For BR and BI, this only holds for $n \geq 10$, but for $n \leq 5$ incremental selection has an advantage regardless of the distance function (see Tab. 5).

Another observation is that although SV5 achieves the highest diversity, it fails at maintaining good "multimodal" indicator values, especially in higher dimensions. SV7 usually obtains slightly better values, except for $k =$ all and $n \geq 5$ (see Fig. 4). Also, the

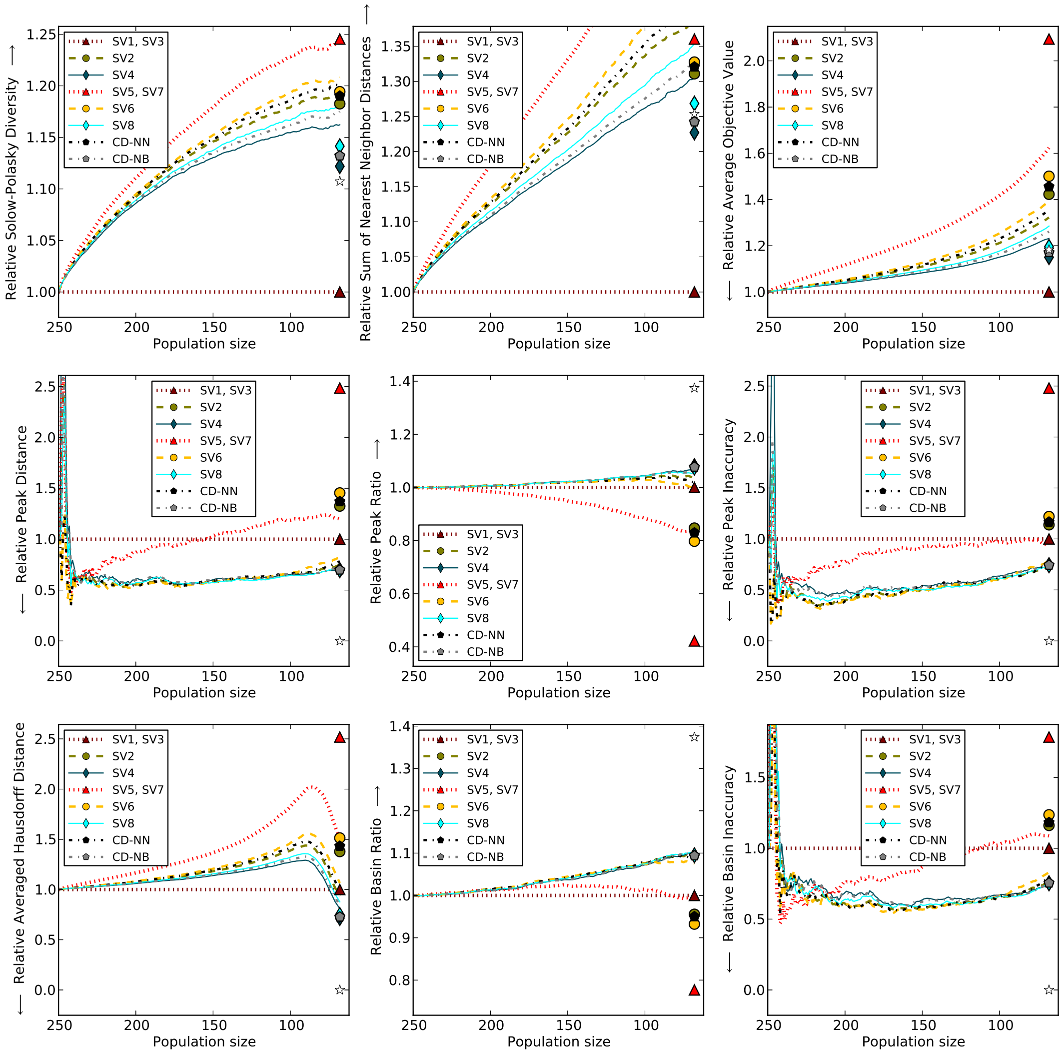

Figure 3: Development of indicator values during incremental selection, averaged over 100 runs. Markers at the right hand side denote the performance of non-incremental versions. The white star stands for the set containing all optima. Arrows at the labels show if the indicator has to be maximized or minimized. SV1 is taken as a reference.

theoretical differences between SPD, SD, and SDNN cannot be recognized in Fig. 3, but they are clearly visible in Fig. 4: For optimizing SPD in low dimensions, it is appropriate to consider only one neighbor. Choosing k = all usually has a bad influence, except on SD and on SPD for $n \geq 5$. For several indicators, $k = 15$ often obtains similarly good values as $k = 1$.

In high dimensions, we can observe an interesting effect of the curse of dimensionality: While the multiobjective selection variants with nearest-better information dominate the

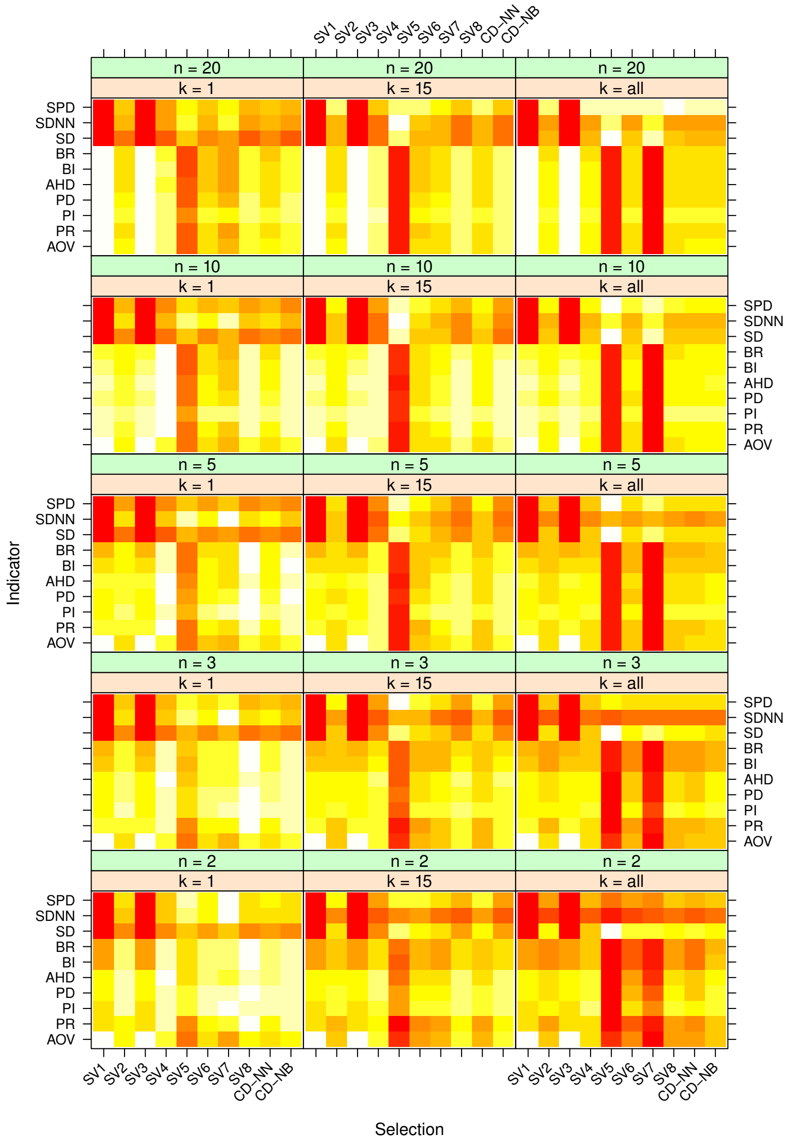

Figure 4: Quality of the representing sets in the subset selection scenario. Lighter shades indicate better values.

competition regarding the multimodal indicators in dimensions up to ten, there seems to be a changeover between ten and 20. In 20 dimensions, suddenly SV1/SV3, which are focusing completely on objective values, are competitive and even seem to obtain the best performance in Fig. 4. This, however, is not true if we regard incremental and non-incremental versions separately. Generally, it looks as if the priority of objective values should increase proportionally with the dimension to obtain good results.

Table 5: Best configurations according to PR, BR, AHD, and BI in experiment 4.2.

n	Sel.	k	Inc.	PR	Std.	p	Sel.	k	Inc.	BR	Std.	p
2	SV8	1	no	0.632	0.084	1	SV6	1	yes	0.734	0.063	1
2	CDNB	1	no	0.627	0.084	0.64	CDNN	1	yes	0.731	0.063	0.62
2	SV4	1	no	0.620	0.087	0.41	SV8	1	yes	0.729	0.072	0.79
3	SV8	1	no	0.727	0.061	1	SV2	1	yes	0.790	0.049	1
3	CDNB	1	no	0.722	0.064	0.56	CDNN	1	yes	0.789	0.046	0.73
3	SV4	1	no	0.719	0.069	0.33	SV8	1	yes	0.788	0.049	0.91
5	SV4	1	no	0.757	0.062	1	CDNB	1	yes	0.794	0.047	1
5	CDNB	1	no	0.756	0.059	0.95	SV8	1	yes	0.792	0.048	0.88
5	SV8	1	no	0.756	0.060	0.90	SV4	1	yes	0.792	0.046	0.81
10	SV4	1	no	0.841	0.050	1	SV4	1	no	0.852	0.044	1
10	CDNB	1	no	0.838	0.049	0.63	CDNB	1	no	0.850	0.042	0.71
10	SV4	15	no	0.835	0.062	0.29	SV8	1	no	0.847	0.042	0.33
20	SV4	15	no	0.932	0.030	1	SV4	15	no	0.934	0.029	1
20	SV1	1	no	0.927	0.042	0.39	CDNB	15	no	0.929	0.028	0.11
20	CDNB	15	no	0.926	0.030	0.10	SV1	1	no	0.927	0.042	0.18

n	Sel.	k	Inc.	AHD	Std.	p	Sel.	k	Inc.	BI	Std.	p
2	CDNB	1	no	0.914	0.341	1	SV6	1	yes	10.13	2.57	1
2	SV8	1	no	0.920	0.370	0.84	CDNN	1	yes	10.20	2.50	0.81
2	SV4	1	no	0.922	0.325	0.66	SV8	1	yes	10.24	2.61	0.72
3	SV8	1	no	1.069	0.285	1	SV2	1	yes	11.92	2.86	1
3	CDNB	1	no	1.093	0.293	0.48	SV8	1	yes	12.02	2.89	0.91
3	SV4	1	no	1.116	0.337	0.29	CDNN	1	yes	12.05	2.86	0.76
5	CDNB	1	no	1.788	0.488	1	CDNB	1	yes	15.55	3.90	1
5	SV4	1	no	1.788	0.524	0.99	SV4	1	yes	15.57	3.83	0.99
5	SV8	1	no	1.795	0.489	0.80	SV8	1	yes	15.69	3.94	0.79
10	SV4	1	no	2.353	0.712	1	SV4	1	no	14.07	4.15	1
10	SV4	15	no	2.385	0.897	0.72	CDNB	1	no	14.29	3.99	0.64
10	CDNB	15	no	2.393	0.862	0.66	SV8	1	no	14.62	4.00	0.24
20	SV4	15	no	1.704	0.750	1	SV4	15	no	6.57	2.87	1
20	SV1	1	no	1.862	1.085	0.29	CDNB	15	no	7.13	2.79	0.11
20	CDNB	15	no	1.862	0.746	0.08	SV1	1	no	7.27	4.23	0.25

Discussion Figure 3 illustrates that regarding diversity and mean objective value, SV1 and SV5 are indeed extreme choices of all selection variants, while SV4 and SV6 are extreme choices of all variants based on non-dominated sorting. We predict that any other selection variant must lie somewhere in between these extreme behaviors. Therefore, this kind of visualization also represents a good baseline investigation for future methods.

Regarding the seeming superiority of SV1/SV3 in 20 dimensions, one explanation might be that the randomly sampled points have a lower probability to be close to an optimum in higher dimensions, and thus also must have comparatively worse objective values than in lower dimensions. Correspondingly, the assumption that all optima are known seems to become increasingly unrealistic with growing dimension. Therefore, this observation should not be seen as an argument against multiobjective selection variants.

The methods investigated here are also highly relevant for the selection problem in drug

Table 6: Additional factors for experiment 4.3

Factor	Type	Symbol	Levels	
			Archive	No Archive
Number peaks	environmental	N	$\{20, 100\}$	
Topology	environmental		{random, linear, funnel}	
Number variables	environmental	n	$\{2, 3, 5, 10, 20\}$	
Incremental	control		{yes, no}	
Number neighbors	control	k	$\{1, 15, \text{all}\}$	
Population size	control	μ	$\{100\}$	$\{100, 500\}$

discovery, described by Meinl et al. [25]. While the run time of their fastest heuristic is only $O(\mu(\mu + \lambda))$, it is not parameterless. The use of SV4 (or similar) may yield better results, albeit with a run time of $O((\mu + \lambda)^2)$.

4.3 Optimization Experiment

Research Question When using a multiobjective evolutionary algorithm for multimodal optimization, which selection variant yields the highest performance? In other words, was the preliminary decision for incremental SV4 without archive and with $k = 1$ in [53] right?

Pre-experimental Planning The assumed situation in this experiment is that of optimization, that is, the selection is now applied repeatedly inside the optimization algorithm. Again, the desired number of individuals surviving the selection, μ, is specified by the user. Regarding archives, the question arises if the decision for or against an archive generally calls for different configurations. At least μ should be chosen smaller to adjust for the run time differences. Additionally, we do test incremental variants together with archives, although the concept seems a bit contradictory in this case. In a sense, it means that we disregard removed points in subsequent selection steps of the current generation, only to re-regard them in following generations by putting them into the archive.

Task All selection variants are compared regarding AHD, an indicator which can sensibly compare sets of different sizes. The performance assessment at the end of each run is solely based on the EA's population and does not consider the archive in any form.

Setup This time, the experiment's setup is determined by the selection variants in Tab. 2 and additional factors from Tab. 6. An evolutionary algorithm is used for multimodal optimization, employing the selection variants as survivor selection. We are using a relatively small budget of $n \cdot 10^3$ objective function evaluations for the EA to keep the run time of the whole experiment endurable. The EA is initialized with a random uniform population. In each generation, $\lambda = 100$ offspring individuals are generated. Mutation is done by adding Gaussian random numbers with a fixed standard deviation of $\sigma = 1$. This is relatively small compared to the size of the search space, which is

$[0, 20]^n$. The test problems are generated the same way as in Sec. 4.2. We are not using any recombination in this experiment, because we suppose that it would either create many noncompetitive offspring or hinder exploration [32]. Violations of the box constraints are repaired by Lamarckian reflection [52]. The archive simply records all individuals that have ever been removed from the population, so its size grows linearly with the number of function evaluations.

Results Figures 5 and 6 show box plots of AHD results in two and 20 dimensions, respectively. In these figures, black dots mark the median of each sample, while the notches depict 95% confidence intervals for the median. By comparing the intervals visually, we can estimate the statistical significance of the differences. Again, the number of peaks and the problem topologies are not accounted for in the visualization, because we expect a certain robustness of the methods against variation of these parameters.

Observations The figures show that using all neighbors is generally a bad idea. This also holds for the dimensions that are not shown here. For multiobjective variants, it seems of minor importance how the non-dominated fronts are sorted: Differences between SV2, SV6, and CD-NN, and between SV4, SV8, and CD-NB are hardly significant. Especially the latter group with nearest-better distances provides a very robust and high performance. Also SV7, the single-objective selection maximizing nearest-better distance, obtains favorable results for $n = 20$, but is sensitive to changes in n, k, and μ. SV7 also generally obtains the best basin ratios (not shown here).

No positive effects could be identified for using an archive. When it is used with non-incremental selection, the results are almost identical to the ones without archive. If incremental selection is used, it even has a negative impact on variants based on nearest-neighbor distances. The comparison of the different population sizes 100 and 500 shows that AHD does not necessarily prefer the larger sets. In 20 dimensions, the indicator values for $\mu = 500$ are comparable to those for $\mu = 100$, in two dimensions the latter choice even attains better results.

Discussion To answer the research question, using an incremental SV4 apparently does not harm, but the non-incremental version yields the same performance with a lower run time. The experiment also confirms the result of experiment 4.2 that conventional nearest-neighbor distances should never be paired with non-incremental selection modes. It is surprising that also the archive has a similarly bad influence on nearest-neighbor distances, and no positive influence anywhere. While one could argue that it may be better suited for global optimization, where only a single solution is sought, this is not the case either. Our results do not indicate any substantial improvement in terms of best or average objective value of the population. It should be investigated if more sophisticated approaches for managing the archive are necessary to obtain any benefit or if this result is due to our way of measuring performance. However, note that in this specific setup, all selection variants except SV5 and SV6 guarantee to retain the global best solution in the population (see Sec. 3). So, it cannot be expected to find a better

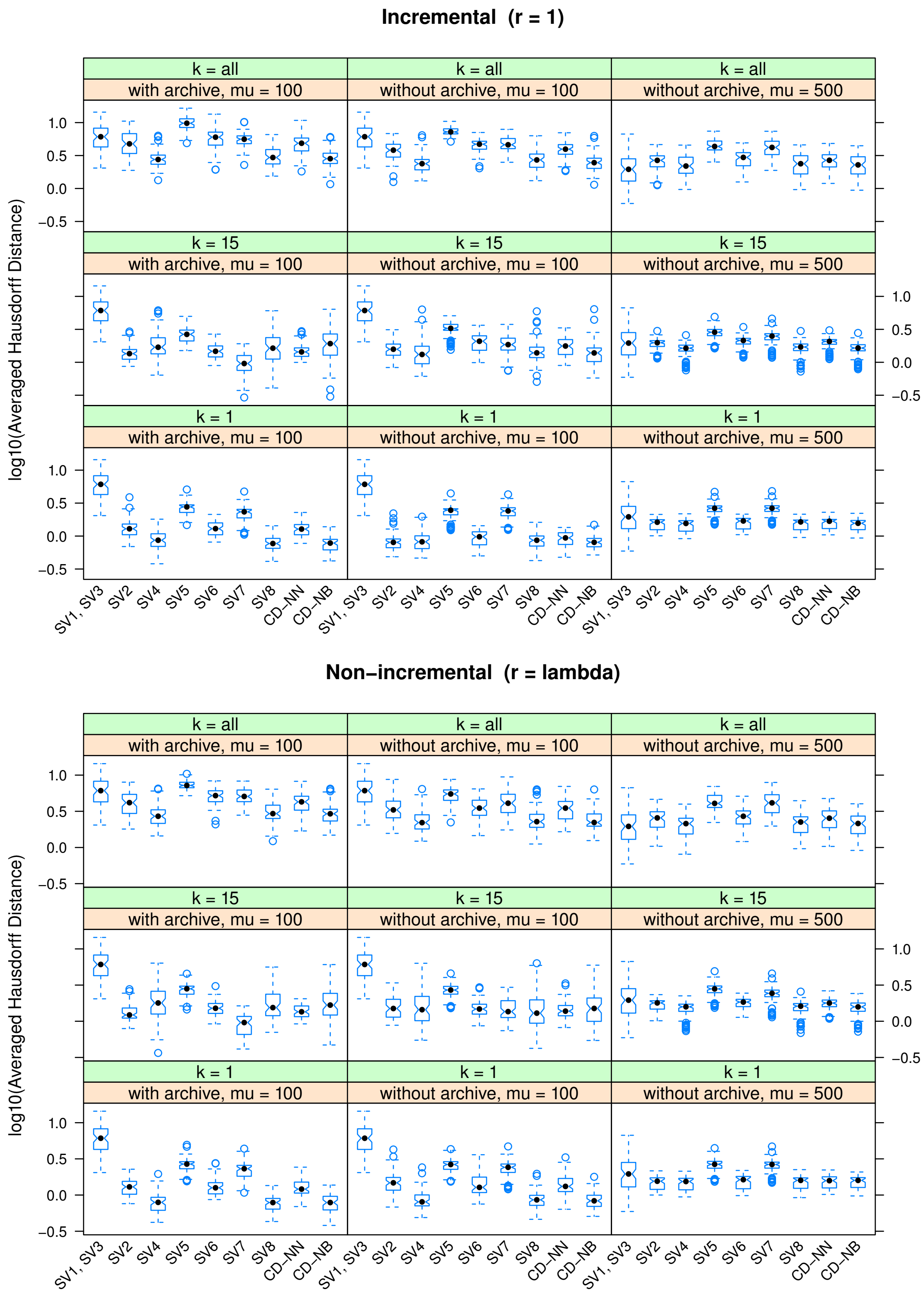

Figure 5: AHD indicator values for $n = 2$ in the optimization scenario.

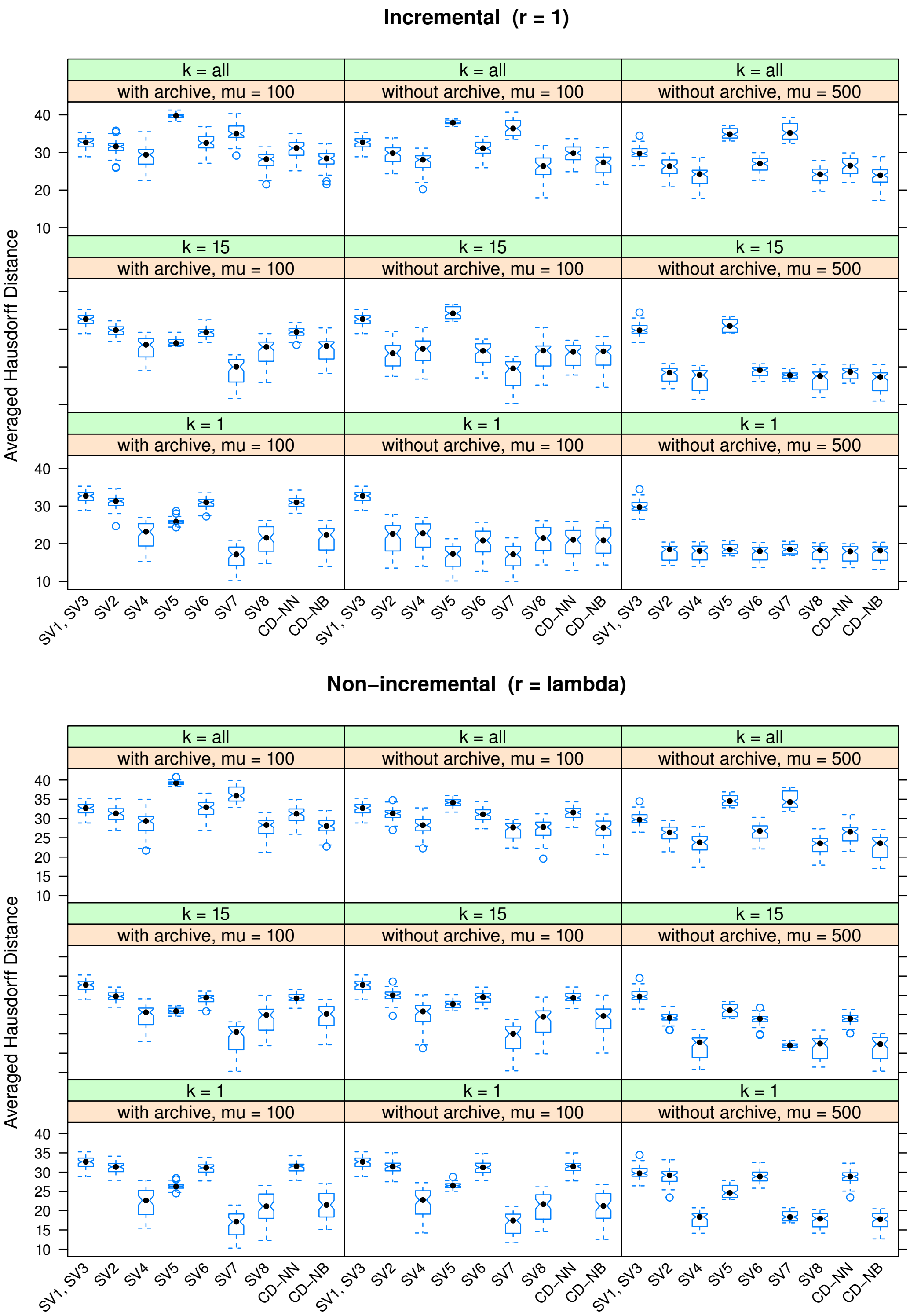

Figure 6: AHD indicator values for $n = 20$ in the optimization scenario.

objective value in the archive than in the population.

Furthermore, only few neighbors should be considered for the distance computations. If we maximize the distance to all neighbors, we simply maximize the spread, but not the diversity of the population, because exterior points are preferred [43, 49]. This simply means a selection pressure towards the corners of the search space. The approach may still be useful for novelty search in a low dimensional, constrained space, but seems inferior for multimodal optimization.

In future experiments, the influence of the optimization budget should be investigated, because also the run length may influence the performance ranking. As none of the regarded selection variants explicitly recognizes basins, it is unlikely (especially for multi-objective variants; see [53]) that the indicator values develop monotonic over time. While in our experiment we could have identified the point in time when the population attains the best performance regarding some indicator, this information currently has no practical relevance, as it is unavailable in real-world situations. We therefore tried to establish a relatively fair comparison by only regarding small budgets.

5 Conclusion

A large amount of experimental data was obtained in this paper and investigated for two different use cases: subset selection and selection during optimization. The former moves on largely uncharted terrain as the problem of actually choosing a suitable subset from the possibly very large result set is currently ignored very often. However, in a real-world scenario, deploying thousands of solutions is not meaningful, as it would be asked too much of a user to perform this selection manually. We provide hints on what indicators and selection methods to use for this. More research on this shall follow. The latter use case (selection during optimization) directly affects the structure of algorithms for multimodal optimization.

The core result of this paper is that the most frequently used multiobjective selection variants for multimodal optimization are not the best available, they can be improved by employing nearest-better distances instead of nearest-neighbor distances. In all our experiments, selection with nearest-better distances yielded the better performance and has been shown to be relatively robust against changes in several other parameters. In particular, it is a great advantage that the super-greedy, non-incremental approach with quadratic run time in the number of solutions (see Sec. 3) can safely be used together with them. Variants with nearest-neighbor distances require an incremental approach with cubic run time to obtain a similar performance. Additionally, in contrast to current practice only a low number of neighbors should be considered, while the criterion applied to sort non-dominated fronts has surprisingly little influence on performance. Our results of course do not rule out that some of the less competitive configurations yield a benefit in different applications than MMO. Further experiments would be especially needed to clarify how archives actually influence the optimization.

Of course, MMO algorithms cannot work by only doing selection. Research on available and new variation operators is in order, and their performance will surely also influence the selection step. This is clearly outside of the focus of this work. However, we presume that locally oriented adaptation of the variation, e. g., basin oriented step size adaptation, or mating restrictions with respect to basin knowledge, would be advantageous. Restricting mates to the nearest neighbors of individuals is already utilized by Qu et al. [34] and Epitropakis et al. [16], and resembles a simple but effective variant of such a mating restriction. Also hybridization with local search may be a practicable way to improve the performance of multiobjective MMO algorithms.

References

[1] Sunith Bandaru and Kalyanmoy Deb. A parameterless-niching-assisted bi-objective approach to multimodal optimization. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 95–102, 2013.

[2] Aniruddha Basak, Swagatam Das, and Kay Chen Tan. Multimodal optimization using a biobjective differential evolution algorithm enhanced with mean distance-based selection. *IEEE Transactions on Evolutionary Computation*, 17(5):666–685, 2013.

[3] David Beasley, David R. Bull, and Ralph R. Martin. A sequential niche technique for multimodal function optimization. *Evolutionary Computation*, 1(2):101–125, June 1993.

[4] Karl Bringmann and Tobias Friedrich. Don't be greedy when calculating hypervolume contributions. In *Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms*, FOGA '09, pages 103–112. ACM, 2009.

[5] Dimo Brockhoff, Tobias Friedrich, Nils Hebbinghaus, Christian Klein, Frank Neumann, and Eckart Zitzler. Do additional objectives make a problem harder? In *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, GECCO '07, pages 765–772. ACM, 2007.

[6] Lam Thu Bui, Hussein A. Abbass, and Jürgen Branke. Multiobjective optimization for dynamic environments. In *IEEE Congress on Evolutionary Computation*, volume 3, pages 2349–2356, 2005.

[7] Carlos A. Coello Coello and Nareli Cruz Cortés. Solving multiobjective optimization problems using an artificial immune system. *Genetic Programming and Evolvable Machines*, 6(2):163–190, 2005.

[8] Emilie Danna and David L. Woodruff. How to select a small set of diverse solutions to mixed integer programming problems. *Operations Research Letters*, 37(4):255–260, 2009.

[9] Swagatam Das, Sayan Maity, Bo-Yang Qu, and Ponnuthurai Nagaratnam Suganthan. Real-parameter evolutionary multimodal optimization – a survey of the state-of-the-art. *Swarm and Evolutionary Computation*, 1(2):71–88, 2011.

[10] Edwin D. de Jong, Richard A. Watson, and Jordan B. Pollack. Reducing bloat and promoting diversity using multi-objective methods. In Lee Spector, editor, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 11–18. Morgan Kaufman, 2001.

[11] Kenneth Alan De Jong. *An analysis of the behavior of a class of genetic adaptive systems.* PhD thesis, University of Michigan, 1975.

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

[13] Kalyanmoy Deb and Amit Saha. Multimodal optimization using a bi-objective evolutionary algorithm. *Evolutionary Computation*, 20(1):27–62, 2012.

[14] Agoston E. Eiben and Mark Jelasity. A critical note on experimental research methodology in EC. In *IEEE Congress on Evolutionary Computation (CEC)*, volume 1, pages 582–587, 2002.

[15] Michael T. M. Emmerich, André H. Deutz, and Johannes W. Kruisselbrink. On quality indicators for black-box level set approximation. In *EVOLVE – A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation*, volume 447 of *Studies in Computational Intelligence*, pages 157–185. Springer, 2013.

[16] Michael G. Epitropakis, Vassilis P. Plagianakos, and Michael N. Vrahatis. Finding multiple global optima exploiting differential evolution's niching capability. In *IEEE Symposium on Differential Evolution (SDE)*, 2011.

[17] Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5):533–549, May 1986.

[18] Julia Handl, Simon C. Lovell, and Joshua Knowles. Multiobjectivization by decomposition of scalar cost functions. In Günter Rudolph, Thomas Jansen, Simon Lucas, Carlo Poloni, and Nicola Beume, editors, *Parallel Problem Solving from Nature – PPSN X*, volume 5199 of *Lecture Notes in Computer Science*, pages 31–40. Springer, 2008.

[19] Saku Kukkonen and Kalyanmoy Deb. Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. In *IEEE Congress on Evolutionary Computation*, pages 1179–1186, 2006.

[20] Joel Lehman and Kenneth O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2):189–223, September 2011.

[21] Joel Lehman, Kenneth O. Stanley, and Risto Miikkulainen. Effective diversity maintenance in deceptive domains. In *Proceeding of the fifteenth annual conference on Genetic and evolutionary computation conference*, GECCO '13, pages 215–222. ACM, 2013.

[22] Jian-Ping Li, Marton E. Balazs, Geoffrey T. Parks, and P. John Clarkson. A species conserving genetic algorithm for multimodal function optimization. *Evolutionary Computation*, 10(3):207–234, 2002.

[23] X. Li, A. Engelbrecht, and M.G. Epitropakis. Benchmark functions for CEC'2013 special session and competition on niching methods for multimodal function optimization. Technical report, RMIT University, Evolutionary Computation and Machine Learning Group, Australia, 2013.

[24] Samir W. Mahfoud. *Niching methods for genetic algorithms*. PhD thesis, University of Illinois at Urbana-Champaign, IL, 1995.

[25] Thorsten Meinl, Claude Ostermann, and Michael R. Berthold. Maximum-score diversity selection for early drug discovery. *Journal of Chemical Information and Modeling*, 51(2):237–247, 2011.

[26] Kaisa Miettinen. Introduction to multiobjective optimization: Noninteractive approaches. In Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen, and Roman Słowiński, editors, *Multiobjective Optimization*, volume 5252 of *Lecture Notes in Computer Science*, pages 1–26. Springer, 2008.

[27] Brad L. Miller and Michael J. Shaw. Genetic algorithms with dynamic niche sharing for multimodal function optimization. In *International Conference on Evolutionary Computation*, pages 786–791, 1996.

[28] Jean-Baptiste Mouret. Novelty-based multiobjectivization. In Stéphane Doncieux, Nicolas Bredèche, and Jean-Baptiste Mouret, editors, *New Horizons in Evolutionary Robotics*, volume 341 of *Studies in Computational Intelligence*, pages 139–154. Springer, 2011.

[29] Alain Pétrowski. A clearing procedure as a niching method for genetic algorithms. In T. Fukuda, T. Furuhashi, and D. B. Fogel, editors, *Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC '96)*, pages 798–803. IEEE Press, 1996.

[30] Mike Preuss and Christian Lasarczyk. On the importance of information speed in structured populations. In *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *Lecture Notes in Computer Science*, pages 91–100. Springer, 2004.

[31] Mike Preuss, Günter Rudolph, and Feelly Tumakaka. Solving multimodal problems via multiobjective techniques with application to phase equilibrium detection. In *IEEE Congress on Evolutionary Computation*, pages 2703–2710, 2007.

[32] Mike Preuss, Lutz Schönemann, and Michael Emmerich. Counteracting genetic drift and disruptive recombination in $(\mu +/, \lambda)$-EA on multimodal fitness landscapes. In *Proceedings of the 2005 conference on Genetic and evolutionary computation*, GECCO '05, pages 865–872. ACM, 2005.

[33] Mike Preuss and Simon Wessing. Measuring multimodal optimization solution sets with a view to multiobjective techniques. In *EVOLVE – A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation IV*, volume 227 of *Advances in Intelligent Systems and Computing*, pages 123–137. Springer, 2013.

[34] Bo-Yang Qu, Ponnuthurai Nagaratnam Suganthan, and Jane-Jing Liang. Differential evolution with neighborhood mutation for multimodal optimization. *IEEE Transactions on Evolutionary Computation*, 16(5):601–614, 2012.

[35] Jani Rönkkönen, Xiaodong Li, Ville Kyrki, and Jouni Lampinen. A generator for multimodal test functions with multiple global optima. In *Simulated Evolution and Learning*, volume 5361 of *Lecture Notes in Computer Science*, pages 239–248. Springer, 2008.

[36] Amit Saha and Kalyanmoy Deb. A bi-criterion approach to multimodal optimization: Self-adaptive approach. In *Simulated Evolution and Learning*, volume 6457 of *Lecture Notes in Computer Science*, pages 95–104. Springer, 2010.

[37] Oliver Schütze, Xavier Esquivel, Adriana Lara, and Carlos A. Coello Coello. Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 16(4):504–522, 2012.

[38] Eduardo Segredo, Carlos Segura, and Coromoto León. Analysing the robustness of multiobjectivisation parameters with large scale optimisation problems. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2012.

[39] Carlos Segura, Carlos A. Coello Coello, Gara Miranda, and Coromoto León. Using multi-objective evolutionary algorithms for single-objective optimization. *4OR*, 11(3):201–228, 2013.

[40] Carlos Segura, Carlos A. Coello Coello, Eduardo Segredo, Gara Miranda, and Coromoto León. Improving the diversity preservation of multi-objective approaches used for single-objective optimization. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 3198–3205, 2013.

[41] Ofer M. Shir. Niching in evolution strategies. In Hans-Georg Beyer, editor, *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*, pages 865–872, New York, NY, USA, 2005. ACM Press.

[42] Ofer M. Shir. Niching in evolutionary algorithms. In Grzegorz Rozenberg, Thomas Bäck, and Joost N. Kok, editors, *Handbook of Natural Computing*, pages 1035–1069. Springer, 2012.

[43] Andrew R. Solow and Stephen Polasky. Measuring biological diversity. *Environmental and Ecological Statistics*, 1(2):95–103, 1994.

[44] Catalin Stoean, Mike Preuss, Ruxandra Stoean, and Dumitru Dumitrescu. Multimodal optimization by means of a topological species conservation algorithm. *IEEE Transactions on Evolutionary Computation*, 14(6):842–864, 2010.

[45] René Thomsen. Multimodal optimization using crowding-based differential evolution. In *IEEE Congress on Evolutionary Computation*, volume 2, pages 1382–1389, 2004.

[46] Andrea Toffolo and Ernesto Benini. Genetic diversity as an objective in multi-objective evolutionary algorithms. *Evolutionary Computation*, 11(2):151–167, 2003.

[47] Aimo Törn, Montaz M. Ali, and Sami Viitanen. Stochastic global optimization: Problem classes and solution techniques. *Journal of Global Optimization*, 14(4):437–447, 1999.

[48] Thanh-Do Tran, Dimo Brockhoff, and Bilel Derbel. Multiobjectivization with NSGA-II on the noiseless BBOB testbed. In *Proceeding of the fifteenth annual conference companion on Genetic and evolutionary computation conference companion*, GECCO '13 Companion, pages 1217–1224. ACM, 2013.

[49] Tamara Ulrich, Johannes Bader, and Lothar Thiele. Defining and optimizing indicator-based diversity measures in multiobjective search. In *Parallel Problem Solving from Nature, PPSN XI*, volume 6238 of *Lecture Notes in Computer Science*, pages 707–717. Springer, 2010.

[50] Tamara Ulrich and Lothar Thiele. Maximizing population diversity in single-objective optimization. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11, pages 641–648. ACM, 2011.

[51] R. K. Ursem. Multinational evolutionary algorithms. In Peter J. Angeline, editor, *Proceedings of the Congress of Evolutionary Computation (CEC-99)*, volume 3, pages 1633–1640, Piscataway, NJ, 1999. IEEE Press.

[52] Simon Wessing. Repair methods for box constraints revisited. In Anna I. Esparcia-Alcázar, editor, *Applications of Evolutionary Computation*, volume 7835 of *Lecture Notes in Computer Science*, pages 469–478. Springer, 2013.

[53] Simon Wessing, Mike Preuss, and Günter Rudolph. Niching by multiobjectivization with neighbor information: Trade-offs and benefits. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 103–110, 2013.

[54] Eckart Zitzler, Joshua Knowles, and Lothar Thiele. Quality assessment of pareto set approximations. In *Multiobjective Optimization*, volume 5252 of *Lecture Notes in Computer Science*, pages 373–404. Springer, 2008.