Efficient Computation of Two-dimensional Solution Sets Maximizing the Epsilon-Indicator

Karl Bringmann
ETH Zurich
Zurich, Switzerland

Tobias Friedrich
Hasso Plattner Institute
Potsdam, Germany

Patrick Klitzke
Universität des Saarlandes
Saarbrücken, Germany

Abstract—**The majority of empirical comparisons of multi-objective evolutionary algorithms (MOEAs) are performed on synthetic benchmark functions. One of the advantages of synthetic test functions is the a-priori knowledge of the optimal Pareto front. This allows measuring the proximity to the optimal front for the solution sets returned by the different MOEAs. Such a comparison is only meaningful if the cardinality of all solution sets is bounded by some fixed k. In order to compare MOEAs to the theoretical optimum achievable with k solutions, we determine best possible ε-indicator values achievable with solution sets of size k, up to an error of δ. We present a new algorithm with runtime $\mathcal{O}(k \cdot \log^2(\delta^{-1}))$, which is an exponential improvement regarding the dependence on the error δ compared to all previous work. We show mathematical correctness of our algorithm and determine optimal solution sets for sets of cardinality $k \in \{2, 3, 4, 5, 10, 20, 50, 100, 1000\}$ for the well known test suits DTLZ, ZDT, WFG and LZ09 up to error $\delta = 10^{-25}$.**

I. Introduction

Biobjective optimization aims at minimizing a fitness function $F\colon \mathcal{X} \to \mathbb{R}^2$ which maps from some arbitrary search space $\mathcal{X}$ to pairs of real numbers, which describe two (potentially conflicting) objective functions F_1 and F_2. This is typically done by finding a solution set $S \subset \mathcal{X}$ such that (i) S approximates well all Pareto optimal points in the objective space $F(\mathcal{X})$ and (ii) S is not too large, say of some fixed size $k = |S|$. We can then compare the solution sets obtained by different biobjective optimization algorithms with a *unary indicator function* $I\colon \mathcal{X}^k \to \mathbb{R}$, which map a solution set to a single real number. Examples for commonly used indicators are the hypervolume indicator, the ε-indicator, generational distance [27], inverted generational distance [11], and several other [29]. The progress of the optimization can then be visualized by plotting this indicator as a function of time (generations or number of fitness evaluations).

For *single-criteria* test functions with known optima it is very common to present a comparison between different optimizers by measuring the *distance to the optimum* of the solutions obtained by the different optimizers. For bi- and multi-objective problems this would mean measuring the difference between the optimal solution set with perfect indicator value and the indicator value of the solution sets obtained by different optimizers. As optimal indicator values are typically hard to obtain for bi- and multi-objective problems, only very few comparisons of multi-objective optimizers present their results with respect to the achieved *distance to the optimum* (e.g. [8, 23]).

What is missing are reliable optimal indicator values for typical test functions, with varying dimension d and solution set size k. For the *hypervolume indicator*, the former website of Eckart Zitzler's group at ETH [30] presents close-to-optimal hypervolume distributions for the ZDT and DTLZ benchmark suites for $d = 2$ and $k = 2, 3, 4, 5, 10, 20, 50, 100, 1000$. They are obtained by the `Maximize` function of Maple's Optimization package and are therefore not guaranteed to be optimal. Last year, Glasmachers [16] obtained slightly larger (=better) hypervolume indicator values than [30] for the same set of test functions and additionally presented fronts for $d = 3$ achieved by a gradient-based optimizer. They are again not proven to be optimal. There are a few theoretical results on optimal hypervolume distributions [3, 4, 9, 15], but obtaining provably-optimal distributions for non-trivial test functions is still an open problem.

We do not consider the hypervolume indicator, but instead study the *ε-indicator* for $d = 2$ dimensions. It measures by how much we have to shift all points of a solution set S in both objectives such that all feasible points in $F(\mathcal{X})$ are dominated. In operations research and theory, this is the most popular measure for multi-criteria optimization [10, 12, 14, 18, 21, 22, 25]. It is also popular in evolutionary multiobjective optimization [6, 19, 26]. The ε-indicator allows us to compare the approximation quality of the solution sets returned by different multi-objective evolutionary algorithms (MOEAs). Our aim is finding the best possible point set of some fixed size k, which maximizes the ε-indicator on a fixed biobjective test function F. More formally, we consider the following problem. We assume that we are given the Pareto front f of the biobjective optimization problem F, for details see Section II.

Optimal ε-indicator problem (OPTEPS): For a given biobjective Pareto front f, solution set size k and error bound δ, determine the best possible ε-indicator achievable with solution sets of size k, up to an additive error of δ.

978-1-4799-7492-4/15/$31.00 ©2015 IEEE

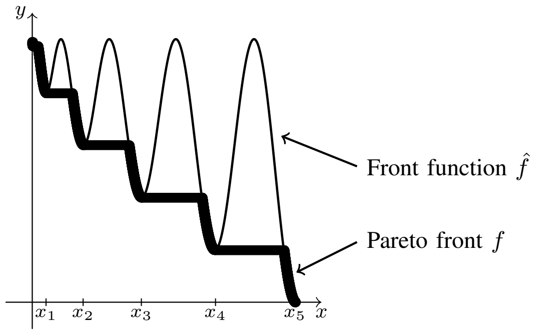


Figure 1: Illustration of our notion of a front function $\hat{f}$ and Pareto front f for the test function WFG2.

We remark that OPTEPS can be solved by picking $n = \Theta(1/\delta)$ equidistant points on the Pareto front and solving the resulting ε-indicator subset selection problem (see [7] for a definition). This implies a runtime of $\mathcal{O}(\delta^{-2}\log(\delta^{-1}))$ [24, 26] or $\mathcal{O}(\delta^{-1}\log(\delta^{-1}))$ [7]. More generally, any algorithm following this approach needs runtime $\Omega(\delta^{-1})$. We present an exponential improvement regarding the dependence on the error δ and present an algorithm with runtime $\mathcal{O}(k\cdot\log^2(\delta^{-1}))$, which makes error bounds of $\delta = 10^{-25}$ and smaller easily tractable.

II. Preliminaries

A. Input Assumptions

We assume that all Pareto optimal points $z \in \mathcal{X}$ have $F_1(z)$, $F_2(z) \in [0,1]$. As input we are given the Pareto front $f\colon [0,1] \to [0,1]$ with

$$f(x) = \inf\{F_2(z) \mid z \in \mathcal{X} \text{ and } F_1(z) \leqslant x\}.$$

Specifically, we assume that we can evaluate $f(x)$ in constant time.

B. Problem Definition

Given a Pareto front f as above and a number k, for any set $P \subset [0,1]$ of size k its ε-indicator is defined as

$$\mathcal{I}_{\text{eps}}(P,f) = \sup_{x\in[0,1]} \min_{p\in P} \max\{p-x, f(p)-f(x)\}.$$

We say that $(p, f(p))$ *ε-approximates* $(x, f(x))$ (or simply *p ε-approximates x*) if $\max\{p-x, f(p)-f(x)\} \leqslant \varepsilon$. We consider the *optimal ε-indicator problem* of computing

$$\text{OPTEPS} := \inf_{\substack{P\subset[0,1]\\|P|=k}} \mathcal{I}_{\text{eps}}(P,f).$$

We remark that OPTEPS is not necessarily attained at some set P, but for any $\varepsilon > 0$ there is a set P with $\mathcal{I}_{\text{eps}}(P,f) \leqslant \text{OPTEPS} + \varepsilon$.

We will solve an approximation version of this problem, where we are additionally given an error parameter δ and want to compute a set $P \subset [0,1]$ of size k with

$$\mathcal{I}_{\text{eps}}(P^*,f) \leqslant \mathcal{I}_{\text{eps}}(P,f) \leqslant \mathcal{I}_{\text{eps}}(P^*,f) + \delta.$$

C. Justification of Input Assumptions

Let us argue why our input assumptions are justified. For typical test problems one can analytically determine a front function $\hat{f}$ describing the best possible objective value F_2 over all points with a fixed objective value F_1, i.e.,

$$\hat{f}(x) = \inf\{F_2(z) \mid z \in \mathcal{X} \text{ and } F_1(z) = x\},$$

and $\hat{f}$ is defined on an interval $[a,b]$ consisting of (a superset of) all possible values of $F_1(z)$, $z \in \mathcal{X}$. If $\hat{f}$ is monotonic decreasing, then it is the desired Pareto front f.

However, a front function $\hat{f}$ might not be monotone, see Figure 1. We argue that then we can derive an approximation $\tilde{f}$ of the true Pareto front f if $\hat{f}$ is sufficiently well-behaved. Assume that (1) $\hat{f}\colon [a,b] \to [c,d]$, (2) $\hat{f}$ has exactly ℓ local minima and we know disjoint intervals $I_1,\dots,I_\ell$ each containing one local minimum, (3) $\hat{f}$ has curvature bounded by C, and (4) a,b,c,d,ℓ, and C are fixed, i.e, we may omit them in $\mathcal{O}$-notation. Let $x_1,\dots,x_\ell$ be the true local minima of $\hat{f}$. (For an illustration see Figure 1.) Then the true Pareto front $f\colon [a,b] \to [c,d]$ of a front function $\hat{f}$ is given by

$$f(x) := \min\{\hat{f}(x), \min\{\hat{f}(x_i) \mid 1 \leqslant i \leqslant \ell \text{ and } x_i \leqslant x\}\}.$$

Thus, we can construct f from $\hat{f}$ if we know $x_1,\dots,x_\ell$. However, our assumptions allow to compute an approximation $\tilde{x}_i \in I_i$ of x_i such that $|\hat{f}(\tilde{x}_i) - \hat{f}(x_i)| \leqslant \delta$ for all i. This can be done using ternary search, which runs in time $\mathcal{O}(\log(\delta^{-1}))$ (using assumptions (3) and (4)). Replacing x_i by $\tilde{x}_i$ in the definition of f yields an approximation $\tilde{f}$ of f such that $f(x) \leqslant \tilde{f}(x) \leqslant f(x) + \delta$ for all x. It can be shown that OPTEPS of $\tilde{f}$ and f differ by at most δ. Moreover, the resulting representation of $\tilde{f}$ allows to evaluate $\tilde{f}(x)$ in constant time (since we assume that ℓ is constant and $\hat{f}$ can be evaluated in constant time).

Finally, after translation and scaling we can assume that $[a,b] = [c,d] = [0,1]$, so that we indeed end up with a Pareto front $\tilde{f}\colon [0,1] \to [0,1]$ that can be evaluated in constant time, as we assume for our algorithm.

Note that the simplification from this section introduces an error of δ, which adds to the error δ of our algorithm presented in the next section. Choosing $\delta := \delta'/2$ yields a total error of $2\delta = \delta'$.

III. Algorithm

A. Discretization

Consider a Pareto front $f\colon [0,1] \to [0,1]$, size k, and error bound δ. Assume without loss of generality that $\delta^{-1} \in \mathbb{N}$, otherwise decrease δ slightly to satisfy this requirement. For approximation purposes, we discretize the domain and range of f to $\Delta := \{0, \delta, 2\delta, \dots, 1-\delta, 1\}$. That is, instead of considering points $x \in [0,1]$ we restrict them to $\Delta \subset [0,1]$, and we let $f^{(\delta)}(x)$ be $f(x)$ rounded up to a multiple of Δ. We obtain a function $f^{(\delta)}\colon \Delta \to \Delta$. This yields an approximate version of the ε-indicator as

$$\mathcal{I}^{(\delta)}_{\text{eps}}(P, f^{(\delta)}) := \max_{x\in\Delta} \min_{p\in P} \max\{p-x, f^{(\delta)}(p) - f^{(\delta)}(x)\}.$$

Now, instead of OPTEPS we will compute

$$\text{APXEPS}^{(\delta)} := \min_{\substack{P \subseteq \Delta \\ |P|=k}} \mathcal{I}_{\text{eps}}^{(\delta)}(P, f^{(\delta)}).$$

Note that APXEPS$^{(\delta)}$ differs from OPTEPS in that we restrict or round all involved objective values to $\Delta \subset [0,1]$. We prove that APXEPS$^{(\delta)}$ is indeed an approximation of OPTEPS.

Theorem 1. *We have* OPTEPS $- \delta \leqslant$ APXEPS$^{(\delta)} \leqslant$ OPTEPS $+ \delta$.

Proof. For the first inequality, let $P_{\text{apx}} \subseteq \Delta$ of size k be a set attaining APXEPS$^{(\delta)}$, i.e.,

$$\text{APXEPS}^{(\delta)} = \mathcal{I}_{\text{eps}}^{(\delta)}(P_{\text{apx}}, f^{(\delta)}).$$

We claim that $\mathcal{I}_{\text{eps}}(P_{\text{apx}}, f) \leqslant \mathcal{I}_{\text{eps}}^{(\delta)}(P_{\text{apx}}, f^{(\delta)}) + \delta$. From this claim and the definition of OPTEPS we obtain the first desired inequality

$$\begin{aligned} \text{OPTEPS} \leqslant \mathcal{I}_{\text{eps}}(P_{\text{apx}}, f) &\leqslant \mathcal{I}_{\text{eps}}^{(\delta)}(P_{\text{apx}}, f^{(\delta)}) + \delta \\ &= \text{APXEPS}^{(\delta)} + \delta. \end{aligned}$$

To show the claim, note that if a point $(x, f^{(\delta)}(x))$, $x \in \Delta$, is ε-approximated by $(p, f^{(\delta)}(p))$, $p \in P_{\text{apx}}$, meaning that $\max\{p - x, f^{(\delta)}(p) - f^{(\delta)}(x)\} \leqslant \varepsilon$, then the point $(x, f(x))$ is $(\varepsilon + \delta)$-approximated by $(p, f(p))$, since $f(p) - f(x) \leqslant f^{(\delta)}(p) - f^{(\delta)}(x) + \delta$. Moreover, for any $x' \in [x - \delta, x]$ we have $p - x' \leqslant p - x + \delta \leqslant \varepsilon + \delta$ and $f(p) - f(x') \leqslant f(p) - f(x) \leqslant \varepsilon + \delta$, so that also $(x', f(x'))$ is ε-approximated by $(p, f(p))$. This yields the claim.

For the second inequality, let $\gamma > 0$ (which we will let go to 0) and consider any set $P_{\text{opt}} \subset [0,1]$ of size k roughly attaining OPTEPS, i.e.,

$$\text{OPTEPS} \leqslant \mathcal{I}_{\text{eps}}(P_{\text{opt}}, f) \leqslant \text{OPTEPS} + \gamma,$$

such a set exists by the definition of OPTEPS as an infimum.

Let P'_{opt} be P_{opt} after rounding up every $x \in P_{\text{opt}}$ to the next multiple of δ. We claim that $\mathcal{I}_{\text{eps}}(P'_{\text{opt}}, f^{(\delta)}) \leqslant \mathcal{I}_{\text{eps}}(P_{\text{opt}}, f) + \delta$. From this claim and the definition of APXEPS$^{(\delta)}$ we obtain

$$\begin{aligned} \text{APXEPS}^{(\delta)} &\leqslant \mathcal{I}_{\text{eps}}^{(\delta)}(P'_{\text{opt}}, f^{(\delta)}) \\ &\leqslant \mathcal{I}_{\text{eps}}(P'_{\text{opt}}, f^{(\delta)}) \\ &\leqslant \mathcal{I}_{\text{eps}}(P_{\text{opt}}, f) + \delta \leqslant \text{OPTEPS} + \delta + \gamma, \end{aligned}$$

where the second inequality follows from the fact that $\mathcal{I}_{\text{eps}}^{(\delta)}$ takes the maximum over a subset of the respective set in $\mathcal{I}_{\text{eps}}$. Letting $\gamma \to 0$ yields the second desired inequality.

To prove the claimed inequality $\mathcal{I}_{\text{eps}}(P'_{\text{opt}}, f^{(\delta)}) \leqslant \mathcal{I}_{\text{eps}}(P_{\text{opt}}, f) + \delta$, consider any point $(x, f(x))$, $x \in [0,1]$, which is ε-approximated by some point $(p, f(p))$, $p \in P_{\text{opt}}$. Let p' be p rounded up to the next multiple of δ. Then $p' - x \leqslant p - x + \delta \leqslant \varepsilon + \delta$. Moreover, by monotonicity of f and the definition of $f^{(\delta)}$, we have $f^{(\delta)}(p') - f^{(\delta)}(x) \leqslant f^{(\delta)}(p) - f^{(\delta)}(x) \leqslant f(p) - f(x) + \delta \leqslant \varepsilon + \delta$. Hence, $(x, f^{(\delta)}(x))$ is $(\varepsilon + \delta)$-approximated by $(p', f^{(\delta)}(p'))$ with $p' \in P'_{\text{opt}}$. This yields the claim. □

We remark that one can compute APXEPS$^{(\delta)}$ by solving the ε-indicator subset selection problem on $\{(x, f^{(\delta)}(x)) \mid x \in \Delta\}$ (with the same reference set). This yields a running time of $\mathcal{O}(\delta^{-1} \log(\delta^{-1}))$ [7]. We design an exponentially faster algorithm with respect to δ^{-1} in the next section.

B. Algorithm Description

We will now present an exact algorithm for APXEPS$^{(\delta)}$, which is also an approximation algorithm for OPTEPS by Lemma 1. It runs in time $\mathcal{O}(k \log^2(\delta^{-1}))$.

Assume that we can check for any given $\varepsilon \in \Delta$ whether APXEPS$^{(\delta)} \leqslant \varepsilon$; we call this check an *ε-test*. From its definition one can see that APXEPS$^{(\delta)}$ is restricted to be a multiple of δ between 0 and 1. Thus, if we can perform an ε-test in time $T(k, \delta)$, then we can exactly compute APXEPS$^{(\delta)}$ by a binary search over $\varepsilon \in \Delta$ in time $\mathcal{O}(\log(|\Delta|)\, T(k, \delta)) = \mathcal{O}(\log(\delta)\, T(k, \delta))$. See Algorithm 1.

Algorithm 1: approximation algorithm for OPTEPS in 2 dimensions, $\mathcal{O}(k \log^2(\delta^{-1}))$ time

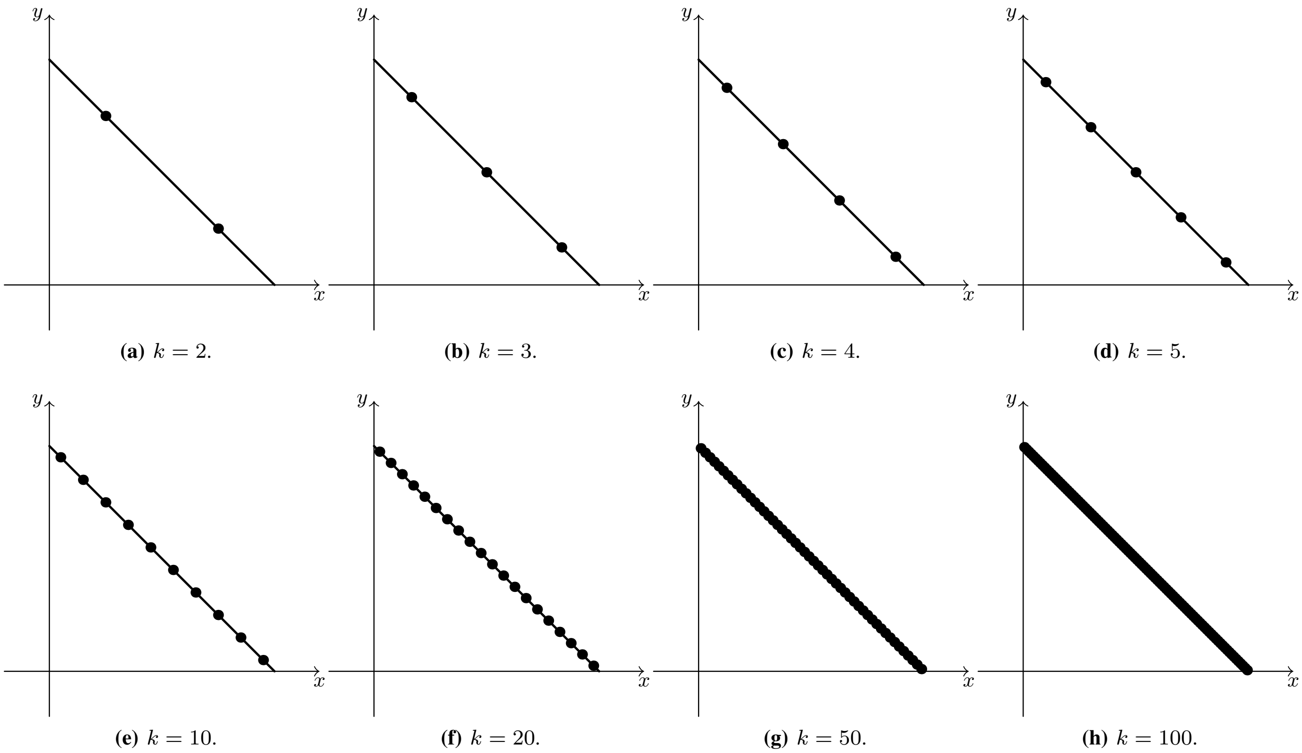
```
Input: integer $k$, Pareto front $f\colon [0,1] \to [0,1]$,
       desired precision $\delta$
Binary search for the minimal $\varepsilon \in \Delta = \{0, \delta, 2\delta, \dots, 1\}$
with APXEPS$^{(\delta)} \geqslant \varepsilon$, which we decide using $\varepsilon$-tests (see
Algorithm 2).
Return $\varepsilon$.
```

Hence, it remains to show that we can perform ε-tests in time $\mathcal{O}(k \log(\delta^{-1}))$. To this end, we use a modified version of the ε-test described in [7]. Using their original description, the discretized front $\{(x, f^{(\delta)}(x)) \mid x \in \Delta\}$ would be explicitly stored, resulting in a time complexity of $\Omega(\delta^{-1})$. Instead, we implicitly store the discretized front, which is possible because the domain Δ is of an easy form and $f^{(\delta)}(x)$ can be evaluated in constant time.

Algorithm 2: ε-test that checks whether APXEPS$^{(\delta)} \leqslant \varepsilon$, $\mathcal{O}(k \log(\delta^{-1}))$ time

```
Input: integer $k$, Pareto front $f\colon [0,1] \to [0,1]$,
       error parameter $\delta$, value $\varepsilon \in \Delta$
$P := \emptyset$
$r := 1$
for $\ell \leftarrow 1$ to $k$ do
    let $x \in \Delta$ be the minimal with $f^{(\delta)}(x) \leqslant f^{(\delta)}(r) + \varepsilon$
    $P := P \cup \{x\}$
    $r := x - \varepsilon - \delta$
    if $r < 0$ then return "APXEPS$^{(\delta)} \leqslant \varepsilon$"
return "APXEPS$^{(\delta)} > \varepsilon$"
```

The algorithm iteratively chooses a set of points P, see Algorithm 2. We denote by $r \in \Delta$ the rightmost point on the front which is not yet ε-approximated by the chosen points. In any iteration, we choose the leftmost point $x \in \Delta$ that can



(a) $k = 2$.

(b) $k = 3$.

(c) $k = 4$.

(d) $k = 5$.

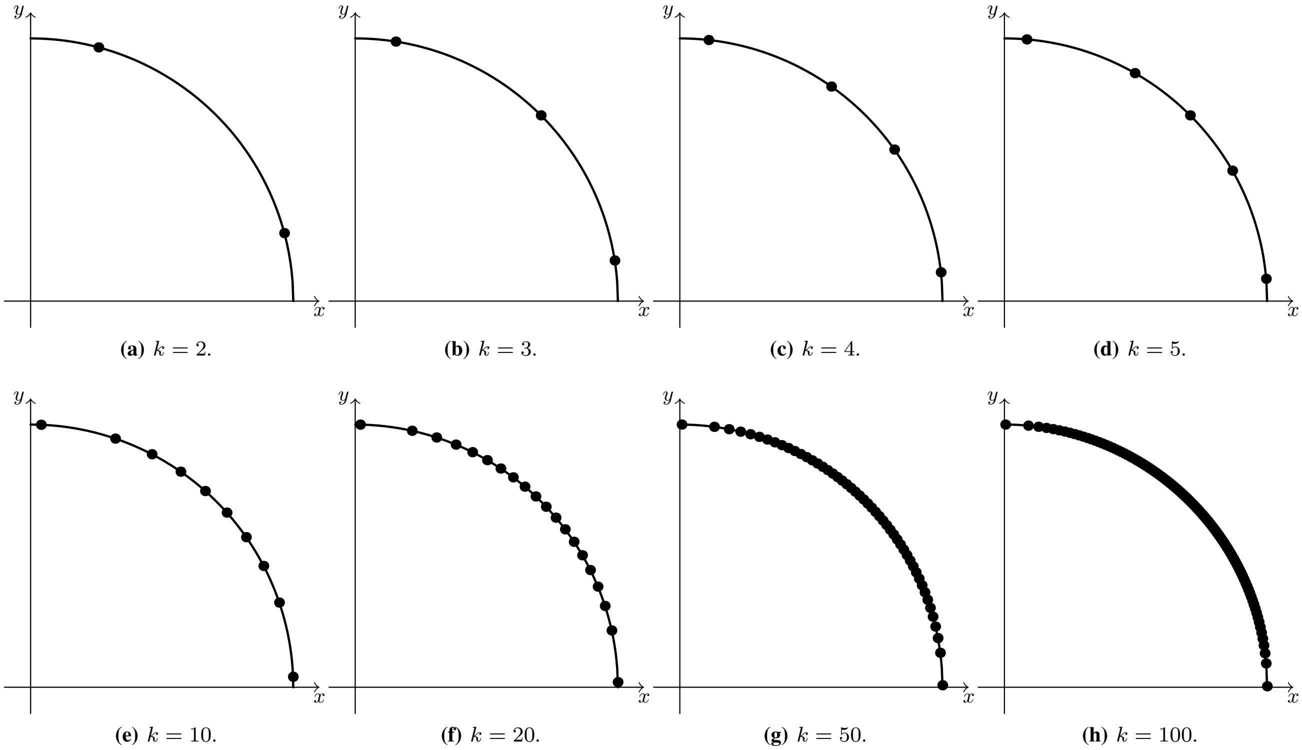
(e) $k = 10$.

(f) $k = 20$.

(g) $k = 50$.

(h) $k = 100$.

Figure 2: ε-optimal point distribution for DTLZ1, WFG3.



(a) $k = 2$.

(b) $k = 3$.

(c) $k = 4$.

(d) $k = 5$.

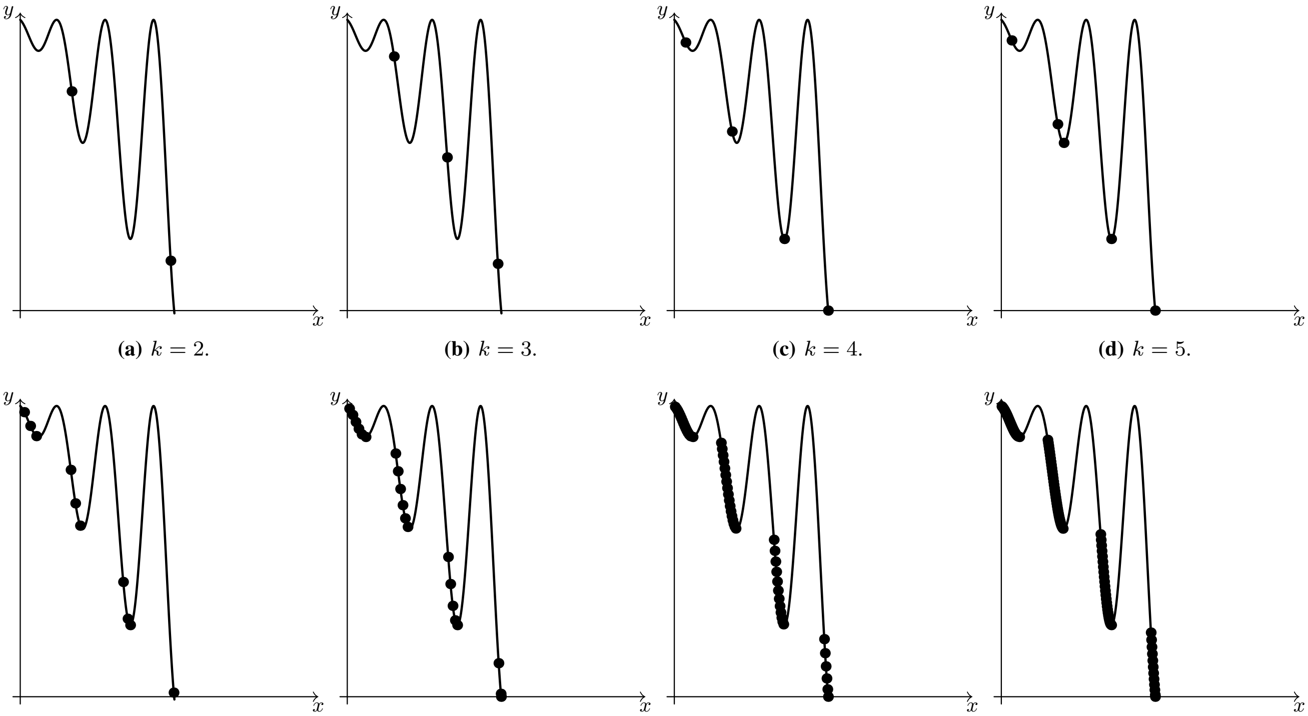
(e) $k = 10$.

(f) $k = 20$.

(g) $k = 50$.

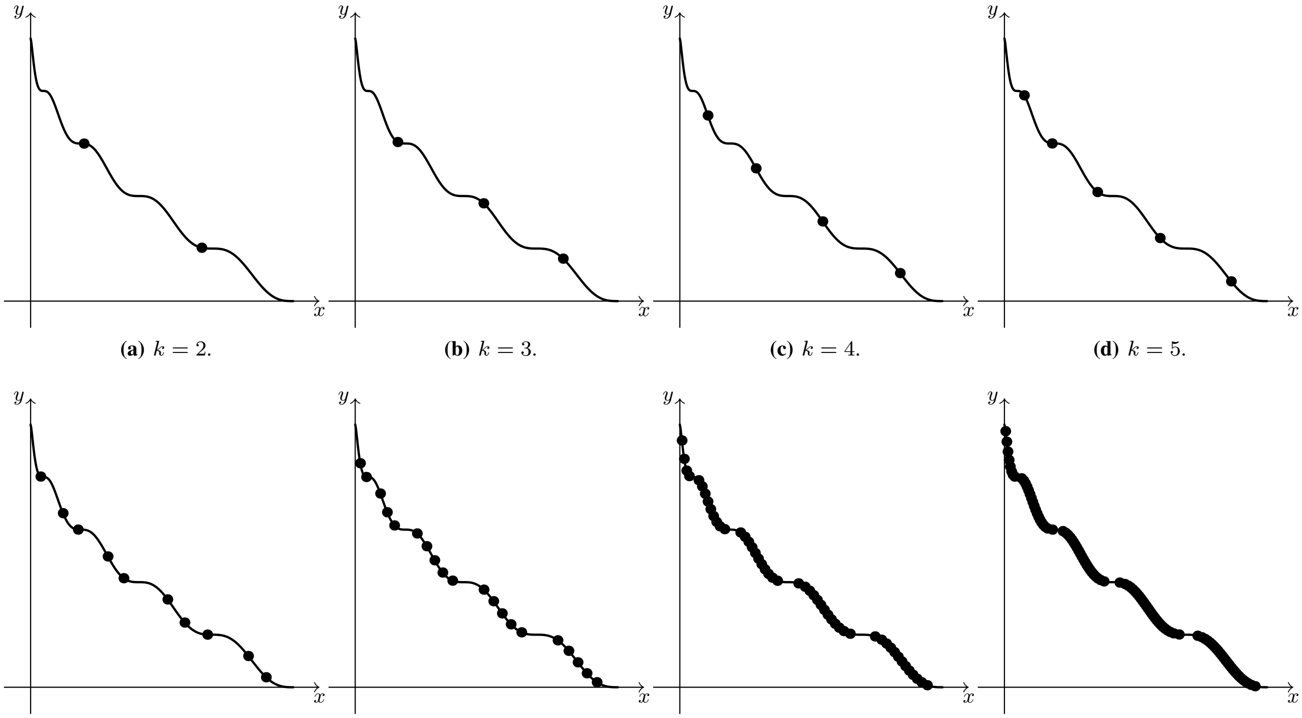
(h) $k = 100$.

Figure 3: ε-optimal point distribution for DTLZ2–4, WFG4–9.



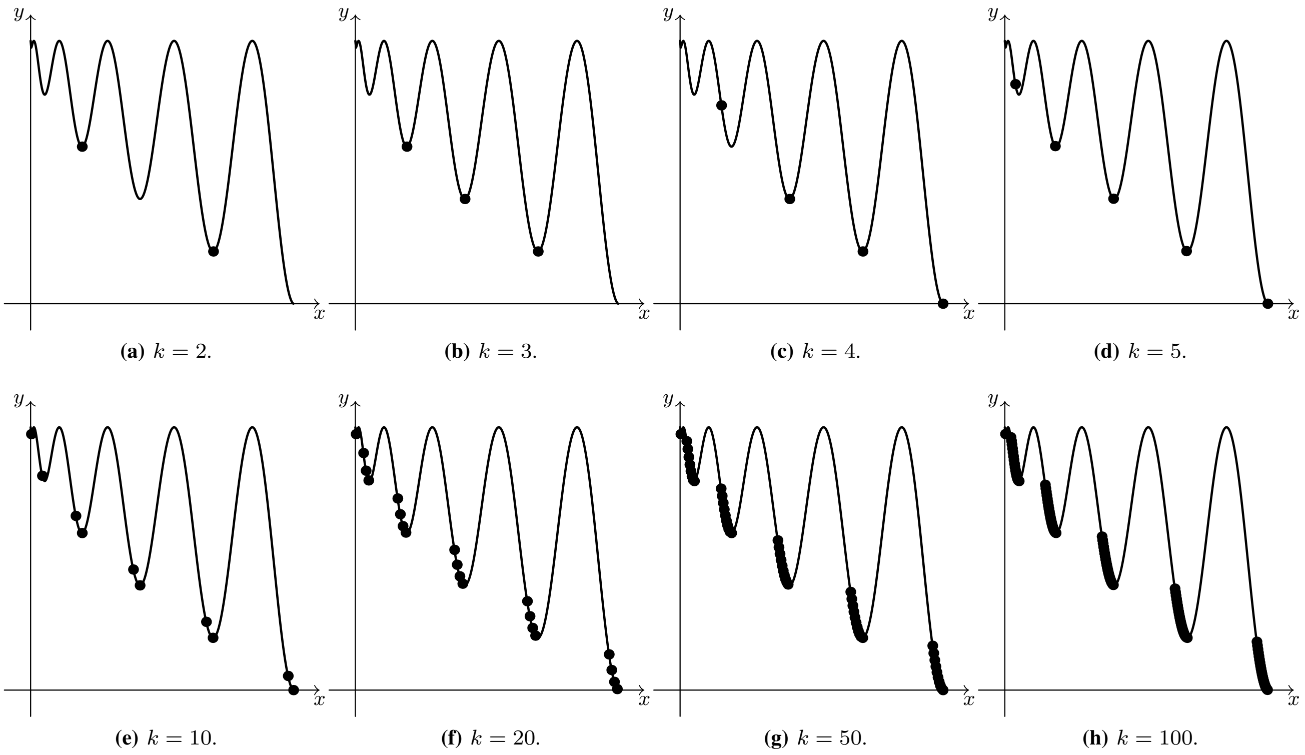
(a) $k = 2$.
(b) $k = 3$.
(c) $k = 4$.
(d) $k = 5$.
(e) $k = 10$.
(f) $k = 20$.
(g) $k = 50$.
(h) $k = 100$.

Figure 4: ε-optimal point distribution for DTLZ7.



(a) $k = 2$.
(b) $k = 3$.
(c) $k = 4$.
(d) $k = 5$.
(e) $k = 10$.
(f) $k = 20$.
(g) $k = 50$.
(h) $k = 100$.

Figure 5: ε-optimal point distribution for WFG1.



(a) $k = 2$.

(b) $k = 3$.

(c) $k = 4$.

(d) $k = 5$.

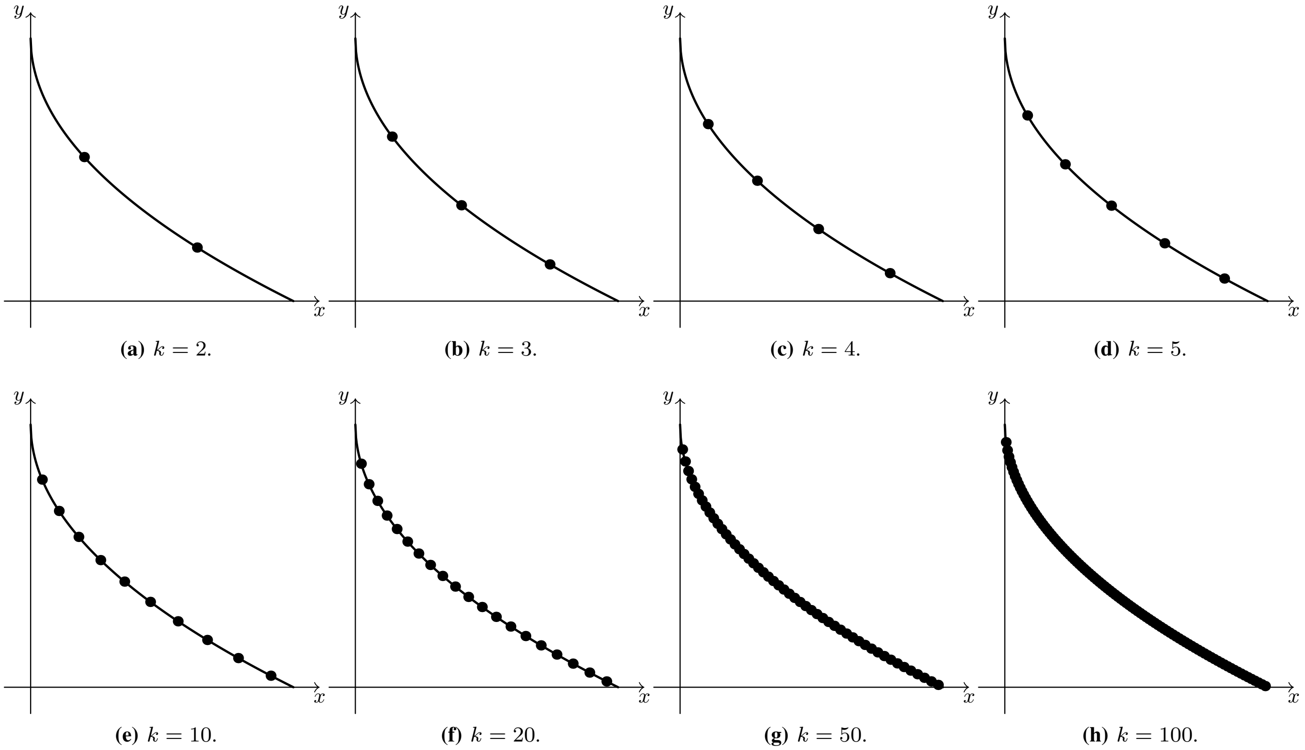
(e) $k = 10$.

(f) $k = 20$.

(g) $k = 50$.

(h) $k = 100$.

Figure 6: ε-optimal point distribution for WFG2.



(a) $k = 2$.

(b) $k = 3$.

(c) $k = 4$.

(d) $k = 5$.

(e) $k = 10$.

(f) $k = 20$.

(g) $k = 50$.

(h) $k = 100$.

Figure 7: ε-optimal point distribution for ZDT1, ZDT4.

ε-approximate r, which amounts to the constraint $f^{(\delta)}(x) \leqslant f^{(\delta)}(r)+\varepsilon$. This point x can be easily found by a binary search over Δ. The algorithm then updates r by deleting all the points that are ε-approximated by the chosen point x, which are all points $y \in \Delta$ with $x-\varepsilon \leqslant y \leqslant r$, i.e., we set $r := x-\varepsilon-\delta$. We repeat this step until all points on the front are ε-approximated by the chosen points, i.e., $r < 0$, or the number of iterations exceeds k, in which case ε was to small.

The correctness is a direct consequence of the ε-test of [7]. Essentially, we argue that some point that ε-approximates $(r, f^{(\delta)}(r))$ has to be chosen, and the point x that we choose is such that it approximates a superset of all other such points, i.e., it is the best possible point approximating $(r, f^{(\delta)}(r))$.

C. Runtime

The binary search for ε in Algorithm 1 takes $\mathcal{O}(\log(\delta^{-1}))$ steps, i.e., $\mathcal{O}(\log(\delta^{-1}))$ invocations of an ε-test. In each loop of the ε-test (Algorithm 2) we perform a binary search taking $\mathcal{O}(\log(|\Delta|)) = \mathcal{O}(\log(\delta^{-1}))$ time. Since the loop is executed at most k times, we obtain an overall time complexity of $\mathcal{O}(k\log(\delta^{-1}))$ for the ε-test and $\mathcal{O}(k\log^2(\delta^{-1}))$ for Algorithm 1.

IV. Experiments

Our experiments have been conducted on a Mac Book Air with 1.7 GHz Intel Core i7 using the GNU Multiple Precision Arithmetic Library [1] and the MPFR C++ library [2]. We approximated the optimal ε-indicator OPTEPS values up to an error of $\delta = 10^{-25}$ for the well known test suits DTLZ [13], ZDT [28], WFG [17] and LZ09 [20] and for solution set sizes $k \in \{2, 3, 4, 5, 10, 20, 50, 100, 1000\}$. Some of the results are shown in Figures 2–7. Note that our plots show the front function $\tilde{f}$ and not the Pareto front f, see Figure 1 for the difference. As a reference for the description of the front functions we used [5]. All results, plots, and the complete source code can be found at http://pklitzke.com/epssubset. Our implementation performed empirically efficient: All aforementioned computations of optimal solution sets for the different fronts and set sizes took together only a a few hours.

V. Conclusions

We present the first efficient algorithm for finding solutions sets of fixed cardinality k which maximize the ε-indicator. Our new algorithm is proven to have an asymptotic runtime of $\mathcal{O}(k\cdot\log^2(\delta^{-1}))$ for an error bound of δ. This is an exponential improvement over all previous algorithms which have to select the optimal subset from equidistantly chosen points and have a runtime of $\Omega(1/\delta)$. We implemented the algorithm, make the source code available, and provide optimal point sets of varying sizes for several common benchmark suites.

Our work enables practitioners to plot the distance to optimum when presenting experiments on EMOAs on these test functions. This allows to judge whether an EMOA eventually gets stuck in local optima or not: If we observe that an EMOA on a test problem F with population size k converges to a value v (i.e., its ε-indicator converges to v), then we can now check whether v is indeed optimal for F and k.

Finding optimal solution sets of bounded cardinality is a notoriously difficult problem. We could only present an efficient algorithm for the ε-indicator in two dimensions. It is an open problem how to obtain similar results in higher dimensions and for other quality measures like the hypervolume indicator. In dimension three and above already the subset selection problem [7, 16, 24, 26] is open for the hypervolume and ε-indicator.

References

[1] The GNU Multiple Precision Arithmetic Library. https://gmplib.org.

[2] MPFR C++ Library. http://www.holoborodko.com/pavel/mpfr/.

[3] A. Auger, J. Bader, and D. Brockhoff. Theoretically investigating optimal μ-distributions for the hypervolume indicator: First results for three objectives. In *11th International Conference on Parallel Problem Solving from Nature (PPSN)*, pages 586–596, 2010.

[4] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler. Hypervolume-based multiobjective optimization: Theoretical foundations and practical implications. *Theor. Comput. Sci.*, 425:75–103, 2012.

[5] J. M. Bader. *Hypervolume-Based Search for Multiobjective Optimization: Theory and Methods*. PhD thesis, Eidgenössische Technische Hochschule ETH Zürich, 2009.

[6] K. Bringmann, T. Friedrich, F. Neumann, and M. Wagner. Approximation-guided evolutionary multi-objective optimization. In *22nd International Joint Conference on Artificial Intelligence (IJCAI '11)*, pages 1198–1203. IJCAI/AAAI, 2011.

[7] K. Bringmann, T. Friedrich, and P. Klitzke. Two-dimensional subset selection for hypervolume and epsilon-indicator. In *16th Annual Conference on Genetic and Evolutionary Computation (GECCO '14)*, pages 589–596. ACM Press, 2014.

[8] K. Bringmann, T. Friedrich, and P. Klitzke. Generic postprocessing via subset selection for hypervolume and epsilon-indicator. In *13th International Conference on Parallel Problem Solving from Nature (PPSN)*, volume 8672 of *LNCS*, pages 518–527. Springer, 2014.

[9] D. Brockhoff. Optimal μ-distributions for the hypervolume indicator for problems with linear bi-objective fronts: Exact and exhaustive results. In *8th International Conference on Simulated Evolution And Learning (SEAL '10)*, volume 6457 of *LNCS*, pages 24–34, 2010.

[10] T. C. E. Cheng, A. Janiak, and M. Y. Kovalyov. Bicriterion single machine scheduling with resource dependent processing times. *SIAM J. on Optimization*, 8:617–630, 1998.

[11] C. A. C. Coello and N. C. Cortés. Solving multiobjective optimization problems using an artificial

immune system. *Genetic Programming and Evolvable Machines*, 6(2):163–190, 2005.

[12] C. Daskalakis, I. Diakonikolas, and M. Yannakakis. How good is the Chord algorithm? In *21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '10)*, pages 978–991, 2010.

[13] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary Multiobjective Optimization*, Advanced Information and Knowledge Processing, pages 105–145. 2005.

[14] I. Diakonikolas and M. Yannakakis. Small approximate Pareto sets for biobjective shortest paths and other problems. *SIAM Journal on Computing*, 39:1340–1371, 2009.

[15] T. Friedrich, C. Horoba, and F. Neumann. Multiplicative approximations and the hypervolume indicator. In *11th Annual Conference on Genetic and Evolutionary Computation (GECCO '09)*, pages 571–578. ACM Press, 2009.

[16] T. Glasmachers. Optimized approximation sets of low-dimensional benchmark pareto fronts. In *13th International Conference on Parallel Problem Solving from Nature (PPSN)*, pages 569–578, 2014.

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

[18] V. Koltun and C. H. Papadimitriou. Approximately dominating representatives. *Theoretical Computer Science*, 371(3):148–154, 2007.

[19] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*, 10(3):263–282, 2002.

[20] H. Li and Q. Zhang. Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II. *IEEE Trans. on Evolutionary Computation*, 13(2):284–302, 2009.

[21] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *41st Annual Symposium on Foundations of Computer Science (FOCS '00)*, pages 86–92. IEEE Press, 2000.

[22] C. H. Papadimitriou and M. Yannakakis. Multiobjective query optimization. In *20th ACM Symposium on Principles of Database Systems (PODS '01)*, pages 52–59, 2001.

[23] M. Pilát and R. Neruda. Hypervolume-based local search in multi-objective evolutionary optimization. In *16th Annual Conference on Genetic and Evolutionary Computation (GECCO '14)*, pages 637–644, 2014.

[24] A. Ponte, L. Paquete, and J. R. Figueira. On beam search for multicriteria combinatorial optimization problems. In *9th International Conference in Integration of AI and OR Techniques in Contraint Programming for Combinatorial Optimzation Problems (CPAIOR '12)*, volume 7298 of *LNCS*, pages 307–321. Springer, 2012.

[25] S. Vassilvitskii and M. Yannakakis. Efficiently computing succinct trade-off curves. *Theor. Comput. Sci.*, 348:334–356, 2005.

[26] D. Vaz, L. Paquete, and A. Ponte. A note on the ε-indicator subset selection. *Theoretical Computer Science*, 499:113–116, 2013.

[27] D. A. V. Veldhuizen. *Multiobjective evolutionary algorithms: Classifications, analyzes, and new innovations*. PhD thesis, Graduate School of Eng. of the Air Force Inst. of Technol., Air Univ., 1999.

[28] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195, 2000.

[29] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evolutionary Computation*, 7:117–132, 2003.

[30] E. Zitzler et al. Density and approximations of μ-distribution for different testproblems, 2009. Computer Engineering and Networks Laboratory (TIK), ETH Zürich, Switzerland, see people.ee.ethz.ch/~sop/download/supplementary/testproblems/.