One-Exact Approximate Pareto Sets for APX-Hard Multiobjective Problems

Fritz Bökler
Institute of Computer Science, Osnabrück University, Germany

Markus Chimani
Institute of Computer Science, Osnabrück University, Germany

Henning Jasper[1]
Institute of Computer Science, Osnabrück University, Germany

Abstract

There are several frameworks to compute *approximate Pareto sets* for multiobjective optimization (MOO) problems. An approximate Pareto set that is even precise in one specific objective is called *one-exact*. In such frameworks, some auxiliary single-objective problem is considered, for which a problem-specific oracle is required. However, often these oracles are required to be a PTAS or even FPTAS. As such, these frameworks are only applicable to "simple" MOO problems that allow for such strong oracles to exist. They are inapplicable whenever the auxiliary problem is **APX**-hard.

We propose a general framework that, for a (possibly even non-constant) accuracy vector $\boldsymbol{\beta} = (\beta_2, \dots, \beta_d)$ and any $\varepsilon > 0$, computes polynomially sized, one-exact $(1, (1+\varepsilon)\beta_2, \dots, (1+\varepsilon)\beta_d)$-Pareto sets for d-objective minimization problems. The framework is analogously applicable to maximization and mixed MOO problems. The running time is polynomial in the time required to solve our auxiliary problem $\boldsymbol{\beta}$-RelaxedDualRestrict. Notably, these guarantees hold even if $\boldsymbol{\beta}$-RelaxedDualRestrict is **APX**-hard. We further show that if $\boldsymbol{\beta}$-RelaxedDualRestrict cannot be solved in polynomial time, then no $(1, \beta_2, \dots, \beta_d)$-Pareto set can be computed in polynomial time. For biobjective problems, our framework even yields a $(1, (1+\varepsilon)\beta_2)$-Pareto set of at most $\mathcal{O}(\log \beta_2)$ times the size of the minimum-size one-exact $(1, (1+\varepsilon)\beta_2)$-Pareto set. We show that this relative size guarantee is asymptotically tight. Further, we present techniques to obtain suitable oracles for $\boldsymbol{\beta}$-RelaxedDualRestrict from existing (single-objective) approximation algorithms, including a general "re-randomization" method that may be of independent interest. Using these, we obtain new best approximation guarantees for several established MOO problems, including Spanner, Clique, TSP, Facility Location, and Set Cover problems.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Mathematical optimization

Keywords and phrases multiobjective optimization, approximate Pareto sets, scalarization

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.112

Funding Supported by the German Research Foundation (DFG), project #517835933.

1 Introduction

Many real-world problems have multiple conflicting goals, e.g., maximize profit with minimum effort. These can be modeled as *multiobjective optimization* (MOO) problems. They enable informed decisions on trade-offs instead of overfitting to a single criterion or linear combination, see [30] for an overview. A MOO problem is a set of instances $(X, \boldsymbol{f})$, where X is (a typically implicit representation of) the set of all feasible solutions and $\boldsymbol{f} = (f_1, \dots, f_d)$ is a vector of d-many objective functions $f_i \colon X \to \mathbb{Q}_{>0}$ for $1 \le i \le d$ (d fixed). W.l.o.g. we can assume

[1] Corresponding author

that all objective functions should be minimized, as maximizing objective f_i is equivalent to minimizing $1/f_i$. Observe that this transformation does not preserve all structural properties, e.g., linearity. Since our arguments are independent of such properties, all of our definitions and results thus analogously hold for maximization objectives. For solutions $x, x' \in X$, we say x *dominates* x' if x is better in at least one objective function and not worse in any of the others. A solution $x \in X$ that is not dominated by any other solution is called *Pareto optimal* (or *efficient*) and its *point* $\boldsymbol{f}(x) = (f_1(x), \ldots, f_d(x))$ is called *non-dominated*.

The *non-dominated set* $\mathcal{Y}^*$ contains all non-dominated points. For many MOO problems, the size of $\mathcal{Y}^*$ can be exponential in the input size, making finding it intractable even if $\mathbf{P} = \mathbf{NP}$. This, among other aspects, motivates the consideration of (polynomial-time) *multiobjective approximation* algorithms[2], see [40] for an overview. A popular concept that closely resembles the one used in single-objective optimization are *approximate Pareto sets*. For $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_d) \in \mathbb{Q}^d_{>0}$, a $\boldsymbol{\mu}$*-Pareto set* $\mathcal{P} \subseteq X$, for any solution $x \in X$, contains a solution $x' \in \mathcal{P}$ that $\boldsymbol{\mu}$*-approximates* x, i.e., $f_i(x') \leq \mu_i \cdot f_i(x)$ for all $1 \leq i \leq d$. If there is one constant $\mu' \in \mathbb{Q}_{>0}$ with $\mu_i \leq \mu'$ for all $1 \leq i \leq d$, they are also called μ'-Pareto sets.

Famously, Papadimitriou and Yannakakis [55] show that, for any $\varepsilon > 0$ and any MOO problem, there exists a $(1+\varepsilon)$-Pareto set of polynomial cardinality $\mathcal{O}((\frac{M}{\varepsilon})^{d-1})$, assuming there exists a polynomial upper bound M on the encoding lengths of the objective function values of feasible solutions [40]. In their corresponding algorithmic framework, M and the objective functions need to be polynomial-time computable. This (very weak) assumption is also necessary in all subsequent frameworks, including ours. They partition the objective space into hyperstrips. For each, they select one solution that approximates all solutions whose points are within this hyperstrip by solving an auxiliary problem called $\textsc{Gap}_{1+\delta}$. They prove that a $(1+\varepsilon)$-Pareto set can be computed in (fully) polynomial time if and only if $\textsc{Gap}_{1+\delta}$ can be solved in (fully) polynomial time for any $\delta > 0$. Here, *fully* polynomial refers to being polynomial in the encoding length of the instance and $1/\delta$. Their method inspired several other frameworks. Herzel et al. [39] are the first (and currently only) that achieve frameworks for one-exact $\boldsymbol{\mu}$-Pareto sets. A $\boldsymbol{\mu}$-Pareto set is *one-exact* if $\mu_1 = 1$, i.e., it is precise w.r.t. the first objective function. Their framework uses $\textsc{DualRestrict}_{1+\delta}$ [24] as an auxiliary problem and, for any $\varepsilon > 0$, yields polynomially sized one-exact $(1+\varepsilon)$-Pareto sets in (fully) polynomial time, given a (fully) polynomial-time subroutine for $\textsc{DualRestrict}_{1+\delta}$ with $1 + \delta = \sqrt{1+\varepsilon}$. For biobjective problems and $1 + \delta = \sqrt[4]{1+\varepsilon}$, they yield one-exact $(1+\varepsilon)$-Pareto sets that are at most twice the size of the smallest such set. Also, note that there are some known problem-specific one-exact approximation algorithms, e.g., for Shortest Path [63] or Scheduling [2] problems.

For $\varepsilon, \delta > 0$, besides $\textsc{Gap}_{1+\delta}$ [34, 55, 64] and $\textsc{DualRestrict}_{1+\delta}$ [24, 39], some frameworks for $(1+\varepsilon)$-Pareto sets use $\textsc{BudgetConstrained}$ [24, 39], $\textsc{Restrict}_{1+\delta}$ [24, 39] or $\textsc{SoftRestrict}_{1+\delta}$ [7] as auxiliary problems to determine suitable solutions for each hyperstrip. Similar to [39], by $1 + \varepsilon$ being some power of the accuracy $1 + \delta$ for which the auxiliary problem is solved, some of them achieve relative size guarantees w.r.t. the smallest $(1+\varepsilon)$-Pareto set [7, 24, 64]. These frameworks require that, for any $\delta > 0$, there is a polynomial-time oracle with accuracy $1 + \delta$ for the respective auxiliary problem. For many MOO problems either no such oracles are known or are unattainable under established complexity assumptions as, e.g., they would solve **APX**-hard problems in polynomial time and thus cannot exist unless $\mathbf{P} = \mathbf{NP}$. Thus, these frameworks are formally inapplicable for many MOO problems. Applying these frameworks anyhow (without proofs), their guarantees

[2] Not to confuse with *multicriteria* (or *bicriteria*) approximation of single-objective problems; see e.g. [23].

may rapidly degrade when oracle accuracy cannot be chosen arbitrarily; e.g., for biobjective problems, the framework of [39] would only yield $(1,(1+\varepsilon)\beta^3)$-Pareto sets for non-choosable problem-specific accuracy β for $\textsc{DualRestrict}_\beta$. To our knowledge, the only frameworks that do not require arbitrary accuracy rely on polynomial-time γ-approximations for the $\textsc{WeightSum}$ [6,8,22,34,37,38] scalarization where $\gamma \geq 1$ may depend on the instance size. Except [6] (which requires further strong assumptions), they are all only applicable to pure minimization problems, as optimum solutions for $\textsc{WeightSum}$ may yield arbitrarily small values in maximization objectives. For $\varepsilon > 0$, for d- and biobjective minimization problems, the best known polynomial-time frameworks achieve $(d\gamma+\varepsilon,\dots,d\gamma+\varepsilon)$-Pareto sets [8] (where at most one worst-case factor is tight), and $((1+2\varepsilon)\gamma,(1+2/\varepsilon)\gamma)$-Pareto sets [37], respectively. In particular, besides being limited to minimization problems, these frameworks do not yield one-exact approximate Pareto sets and exhibit no relative size guarantee w.r.t. the minimum-size approximate Pareto set of the respective quality.

Contribution. We propose a general framework to compute polynomially-sized one-exact $(1,(1+\varepsilon)\beta_2,\dots,(1+\varepsilon)\beta_d)$-Pareto sets $\mathcal{P}$ for d-objective minimization problems, for a (possibly non-constant) accuracy vector $\boldsymbol{\beta}=(\beta_2,\dots,\beta_d)$ and any $\varepsilon>0$; see Section 3 for the precise formalisms. The framework is also applicable to maximization and mixed optimization problems, replacing $1+\varepsilon$ by $1-\varepsilon$ accordingly. The running time is polynomial in the time required to solve our auxiliary problem $\boldsymbol{\beta}$-$\textsc{RelaxedDualRestrict}$. This can be done in polynomial time for many interesting MOO problems for which all other existing frameworks are inapplicable. We further show that if $\boldsymbol{\beta}$-$\textsc{RelaxedDualRestrict}$ cannot be solved in polynomial time, then no $(1,\beta_2,\dots,\beta_d)$-Pareto set can be computed in polynomial time.

For biobjective problems, where $\boldsymbol{\beta}=(\beta_2)$, our framework yields a set of at most $\mathcal{O}(\log\beta_2)$ times the size of the minimum-size one-exact $(1,(1+\varepsilon)\beta_2)$-Pareto set. We show that this guarantee is asymptotically tight. Further, we present simple techniques to obtain suitable oracles for $\boldsymbol{\beta}$-$\textsc{RelaxedDualRestrict}$ from existing approximation algorithms, including a general "re-randomization" method that may be of independent interest. We give a detailed discussion on how our framework can be applied in the context of the Multiobjective Spanner problem [16]. We conclude with several more established MOO problems for which our framework achieves new best approximation guarantees. Some of these are best-possible up to a factor of $1+\gamma$, for any constant $\gamma>0$, under standard complexity assumptions.

2 Preliminaries

For $n,c\in\mathbb{N}_{\geq 1}$, let $[n]\coloneqq\{1,\dots,n\}$ and $[n]_{\geq c}\coloneqq\{c,\dots,n\}$. As discussed above, we assume w.l.o.g. that all objective functions should be minimized. Let $d\in\mathbb{N}_{\geq 1}$. In a d-objective minimization problem instance $(X,\boldsymbol{f})$, for solutions $x,x'\in X$, x dominates x' if $f_j(x)<f_j(x')$ for at least one $j\in[d]$ and $f_i(x)\leq f_i(x')$ for all $i\in[d]$. For a vector $\boldsymbol{\mu}=(\mu_1,\mu_2,\dots,\mu_d)\in\mathbb{Q}^d_{\geq 1}$, x $\boldsymbol{\mu}$-approximates x' if $f_i(x)\leq\mu_i\cdot f_i(x')$ for all $i\in[d]$. A set of feasible solutions $\mathcal{P}\subseteq X$ is called a $\boldsymbol{\mu}$-Pareto set if for every solution $x'\in X$ there is an $x\in\mathcal{P}$ that $\boldsymbol{\mu}$-approximates x'. If, for some $\mu'\in\mathbb{Q}_{\geq 1}$, $\mu_i\leq\mu'$ for all $i\in[d]$ we call it a μ'-Pareto set. For $\varepsilon>0$, in literature [40] $(1+\varepsilon,\dots,1+\varepsilon)$-Pareto sets are often called ε-Pareto sets; for consistency, we call them $(1+\varepsilon)$-Pareto sets. For $\boldsymbol{\mu}=(1,\mu',\dots,\mu')\in\mathbb{Q}^d_{\geq 1}$, we use the shorthand *one-exact* μ'-Pareto set. An algorithm computing a $\boldsymbol{\mu}$-Pareto set is a $\boldsymbol{\mu}$-approximation. For $j\in[d]$ and budget vector $\boldsymbol{b}\in\mathbb{Q}^{d-1}_{>0}$, with entries b_i for all $i\in[d]\setminus\{j\}$, let $\mathrm{OPT}_j(\boldsymbol{b})\coloneqq\min_{x\in X}\{f_j(x)\mid f_i(x)\leq b_i,\forall i\in[d]\setminus\{j\}\}$ be the minimum objective function value in f_j under budgets b_i for all other objectives (or $+\infty$ if no such solution exists). With $\langle I\rangle$, we denote the binary encoding length of an input I. For $i\in[d]$, $f_i(X)$ denotes the image of function f_i.

For $\boldsymbol{\beta'} = (\beta'_2, \ldots, \beta'_d) \in \mathbb{Q}^{d-1}_{\geq 1}$ and $\varepsilon > 0$, we make some observations on the size of one-exact $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto sets. For any fixed $d \in \mathbb{N}_{\geq 1}$, there are instances of MOO problems with d objectives where the minimum-size one-exact $(1+\varepsilon)$-Pareto sets can be arbitrarily larger than the minimum-size (not one-exact) $(1+\varepsilon)$-Pareto set [39]. An analogous construction gives us Observation 1.

▶ **Observation 1.** *For any $\varepsilon > 0$, $\boldsymbol{\beta'} \in \mathbb{Q}^{d-1}_{\geq 1}$, and any $N \in \mathbb{N}_{\geq 1}$ there are instances of MOO problems with d objectives with minimum-size one-exact $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto set $\mathcal{P}^{\min}$ and minimum-size (not one-exact) $(1+\varepsilon)$-Pareto set $\mathcal{P}^{\min}_{1+\varepsilon}$ such that $|\mathcal{P}^{\min}| > N \cdot \mathcal{P}^{\min}_{1+\varepsilon}$.*

However, we can also show that the minimum-size $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto set can be smaller than both the minimum-size one-exact $(1+\varepsilon)$-, and (not one-exact) $(1+\varepsilon)$-Pareto set.

▶ **Lemma 2.** *For biobjective minimization problems, for any $\delta > \varepsilon > 0$ and $\beta'_2 \in \mathbb{Q}_{>1}$, a minimum-size $(1, (1+\varepsilon)\beta'_2)$-Pareto set $\mathcal{P}^{\min}$ can be factor $\lfloor \log_{1+\delta}((1+\varepsilon)\beta'_2) \rfloor + 1$ smaller than both a minimum-size $(1, 1+\varepsilon)$-Pareto set $\mathcal{P}^{\min}_1$ and a minimum-size $(1+\varepsilon, 1+\varepsilon)$-Pareto set $\mathcal{P}^{\min}_{1+\varepsilon}$.*

Proof. Let $X = \{x^{(1)}, x^{(2)}, \ldots, x^{(k+1)}\}$ with $k = \lfloor \log_{1+\delta}((1+\varepsilon)\beta'_2) \rfloor$ and $f_1(x^{(i)}) = (1+\delta)^{i-1}$ and $f_2(x^{(i)}) = \frac{(1+\varepsilon)\beta'_2}{(1+\delta)^{i-1}}$. There are no two solutions $x, x' \in X$ such that x $(1, 1+\varepsilon)$-approximates (or $(1+\varepsilon, 1+\varepsilon)$-approximates) x'. Thus, $\mathcal{P}^{\min}_1 = \mathcal{P}^{\min}_{1+\varepsilon} = X$ and $|\mathcal{P}^{\min}_1| = |\mathcal{P}^{\min}_{1+\varepsilon}| = k+1$. In contrast, $x^{(1)}$ $(1, (1+\varepsilon)\beta'_2)$-approximates all other $x' \in X$ and we get $\mathcal{P} = \{x^{(1)}\}$ and $|\mathcal{P}| = 1$. ◀

3 Frameworks for One-Exact Approximate Pareto Sets

In this section, we present our approximation frameworks in the context of minimization. All definitions and arguments made in this section can analogously be adapted to maximization or mixed MOO problems. First, we formally define our auxiliary problem $\boldsymbol{\beta}$-RelaxedDualRestrict and give a simple framework to yield one-exact $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-Pareto sets for multiobjective minimization problems. Afterwards, we give a slightly more elaborate framework for biobjective minimization problems.

▶ **Definition 3** ($\boldsymbol{\beta}$-RelaxedDualRestrict)**.** *Let $\mathcal{I}$ be the set of instances of a d-objective minimization problem. Let $\boldsymbol{\beta} = (\beta_2, \ldots, \beta_d)$ be an accuracy vector, with polynomial-time computable $\beta_i \colon \mathcal{I} \to \mathbb{Q}_{\geq 1}$ for all $i \in [d]_{\geq 2}$. Given an instance $I = (X, \boldsymbol{f}) \in \mathcal{I}$ and a budget vector $\boldsymbol{b} = (b_2, \ldots, b_d) \in \mathbb{Q}^{d-1}_{>0}$, the auxiliary problem $\boldsymbol{\beta}$-RelaxedDualRestrict asks to return an $x \in X$ with $f_i(x) \leq \beta_i(I) \cdot b_i$ for all $i \in [d]_{\geq 2}$ and $f_1(x) \leq OPT_1(\boldsymbol{b})$; alternatively, if there is no feasible solution $x' \in X$ with $f_i(x') \leq b_i$ for all $i \in [d]_{\geq 2}$, we may return $x = NO$. W.l.o.g., we can assume that an oracle answers NO in case of a budget below 2^{-M} (the minimum objective function value of any feasible solution).*

The β-functions model instance-dependent approximation ratios; often individual β-functions may be constant. This is a stark contrast to DualRestrict$_{1+\delta}$ proposed in [24], where an approximation scheme needs to exist for arbitrary small accuracy $1+\delta$. Hence, the class of MOO problems for which $\boldsymbol{\beta}$-RelaxedDualRestrict can be solved in polynomial time contains the class for which DualRestrict$_{1+\delta}$ is solvable in fully polynomial time. Notably, it also includes all problems for which DualRestrict$_{1+\delta}$ is **APX**-hard; there, the frameworks [24, 39] based on DualRestrict$_{1+\delta}$ are not applicable. Existing auxiliary problems (as, e.g., DualRestrict$_{1+\delta}$), typically use the same accuracy $1+\delta$

in all objectives: they all need to be approximable arbitrarily well anyways. In contrast, $\boldsymbol{\beta}$-RelaxedDualRestrict allows separate arbitrary accuracies $\beta_i(I)$ for each objective f_i, $i \in [d]_{\geq 2}$. In the following, we do not explicitly state the instance I as part of the input of $\boldsymbol{\beta}$ or $\boldsymbol{\beta}$-RelaxedDualRestrict as this should always be clear from the context.

While some methods assuming arbitrarily accurate oracles can be adapted to our setting, many are too lavish when accuracy is limited: for example for biobjective problems, the extremely fine-grained local search of the solution space by Herzel et al. [39] would amplify the (in)accuracy β_2 by a power of 3 – only yielding $(1, (1+\varepsilon)(\beta_2)^3)$-Pareto sets. Hence, our main technical challenges lie in selecting solutions that are guaranteed to approximate many other solutions, while we have so little insight into the solution space. We also show that our methods achieve asymptotically best-possible guarantees.

3.1 Multiobjective Problems

For any d-objective optimization problem there is a one-exact $(1+\varepsilon)$-Pareto set of size $\mathcal{O}((\frac{M}{\log(1+\varepsilon)})^{d-1})$, for any $\varepsilon > 0$ [39]. As a warm-up, we show that there exists an even smaller one-exact $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto set for any $\varepsilon > 0$ and $\boldsymbol{\beta'} \in \mathbb{Q}^{d-1}_{\geq 1}$. Note that 2^M and 2^{-M} bound the largest and smallest occurring objective function values, respectively.

▶ **Lemma 4.** *For $\varepsilon > 0$ and $\boldsymbol{\beta'} \in \mathbb{Q}^{d-1}_{\geq 1}$, every d-objective minimization instance $(X, \boldsymbol{f})$ admits a one-exact $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto set of size at most $\mathcal{O}((\frac{M}{\log((1+\varepsilon)\beta'_{\min})})^{d-1})$, for $\beta'_{\min} := \min_{i \in [d]_{\geq 2}} \beta'_i$.*

Proof. We partition the entire objective space $[2^{-M}, 2^M] \times \cdots \times [2^{-M}, 2^M]$ into hyperstrips $[2^{-M}, 2^M] \times [((1+\varepsilon)\beta'_2)^{j_2}, ((1+\varepsilon)\beta'_2)^{j_2+1}] \times \cdots \times [((1+\varepsilon)\beta'_d)^{j_d}, ((1+\varepsilon)\beta'_d)^{j_d+1}]$, with $j_i \in \{-\lceil \frac{M \log 2}{\log((1+\varepsilon)\beta'_i)} \rceil, \ldots, -1, 0, 1, \ldots, \lceil \frac{M \log 2}{\log((1+\varepsilon)\beta'_i)} \rceil - 1\}$, for each $i \in [d]_{\geq 2}$. This induces $\prod_{i=2}^d (2 \cdot \lceil \frac{M \log 2}{\log((1+\varepsilon)\beta'_i)} \rceil) \leq (2 \cdot \lceil \frac{M \log 2}{\log((1+\varepsilon)\beta'_{\min})} \rceil)^{d-1}$ many hyperstrips. For each non-empty hyperstrip H, we add one representative solution $x \in X$ to the set $\mathcal{P}$ with $\boldsymbol{f}(x) \in H$ and $f_1(x) \leq f_1(x')$ for all $x' \in X$ with $\boldsymbol{f}(x') \in H$. Then x $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-approximates all other solutions whose points are in H. Thus, $\mathcal{P}$ is a $(1, (1+\varepsilon)\beta'_2, \ldots, (1+\varepsilon)\beta'_d)$-Pareto set of size $|\mathcal{P}| \in \mathcal{O}((\frac{M}{\log((1+\varepsilon)\beta'_{\min})})^{d-1})$. ◀

Leveraging Lemma 4 to algorithmically construct an approximate Pareto set of this size requires to determine a suitable representative for every hyperstrip. However, due to the possibly non-constant (in-)accuracies $\boldsymbol{\beta}$, directly using $\boldsymbol{\beta}$-RelaxedDualRestrict does not work, as the strips are too wide. However, for narrower hyperstrips – whose upper and lower bound are only a factor $(1+\varepsilon)$ apart in every dimension $i \in [d]_{\geq 2}$ – $\boldsymbol{\beta}$-RelaxedDualRestrict does indeed yield representative solutions.

▶ **Lemma 5.** *For $\varepsilon > 0$, $\boldsymbol{b} \in \mathbb{Q}^{d-1}_{>0}$, and hyperstrip $H = [2^{-M}, 2^M] \times [\frac{1}{(1+\varepsilon)} b_2, b_2] \times \cdots \times [\frac{1}{(1+\varepsilon)} b_d, b_d]$, any solution $x \in X$ to $\boldsymbol{\beta}$-RelaxedDualRestrict$(\boldsymbol{b})$ $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-approximates every $z \in X$ with $\boldsymbol{f}(z) \in H$.*

Proof. If $\boldsymbol{\beta}$-RelaxedDualRestrict$(\boldsymbol{b})$ yields NO, H is empty. Otherwise H is non-empty and let $x \in X$ be the returned solution. Then $f_1(x) \leq \mathrm{OPT}_1(\boldsymbol{b})$, so there is no $x' \in X$ with $f_1(x') < f_1(x)$ and $f_i(x') \leq b_i$ for all $i \in [d]_{\geq 2}$. Further, $f_i(x) \leq \beta_i b_i$ for all $i \in [d]_{\geq 2}$. Let $z \in X$ with $\boldsymbol{f}(z) \in H$. Since, $f_i(z) \leq b_i$ for all $i \in [d]_{\geq 2}$, we have $f_1(z) \geq f_1(x)$. Also, since $f_i(z) \geq \frac{1}{(1+\varepsilon)} b_i$, we get $(1+\varepsilon)\beta_i f_i(z) \geq (1+\varepsilon)\beta_i \cdot \frac{1}{(1+\varepsilon)} \cdot b_i = \beta_i b_i \geq f_i(x)$. Thus, z is $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-approximated by x. ◀

Algorithm 1 A $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-approximation for d-objective minimization problems.

Input: instance $(X, \boldsymbol{f})$ of a d-objective minimization problem, $\varepsilon > 0$

$\mathcal{P} \leftarrow \emptyset$
for all $(j_2, \ldots, j_d)$ s.t. $j_i \in \{-\lceil \frac{M \log 2}{\log(1+\varepsilon)} \rceil, \ldots, \lceil \frac{M \log 2}{\log(1+\varepsilon)} \rceil - 1\}$ for each $i \in [d]_{\geq 2}$ **do**
  $\boldsymbol{b} \leftarrow ((1+\varepsilon)^{j_2}, \ldots, (1+\varepsilon)^{j_d})$
  $x \leftarrow$ solve $\boldsymbol{\beta}$-RelaxedDualRestrict$(\boldsymbol{b})$
  if $x \neq$ NO **then** $\mathcal{P} \leftarrow \mathcal{P} \cup \{x\}$
return $\mathcal{P}$

Note that the point $\boldsymbol{f}(x)$ above does not necessarily lie within the hyperstrip H. Consider hyperstrips of width $(1+\varepsilon)^{j_i}$ with $j_i \in \{-\lceil \frac{M \log 2}{\log(1+\varepsilon)} \rceil, \ldots, -1, 0, 1, \ldots, \lceil \frac{M \log 2}{\log(1+\varepsilon)} \rceil - 1\}$, for each dimension $i \in [d]_{\geq 2}$. These hyperstrips satisfy the condition of Lemma 5. Now, our framework, as shown in Algorithm 1, only needs to iterate over all of these hyperstrips. If a hyperstrip is non-empty, we add a solution of the respective $\boldsymbol{\beta}$-RelaxedDualRestrict instance to $\mathcal{P}$. By Lemma 5, every such solution $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-approximates every solution whose point is in that hyperstrip. Thus, we yield:

▶ **Theorem 6.** *For $\varepsilon > 0$ and any instance $I = (X, \boldsymbol{f})$ of a d-objective minimization problem, Algorithm 1 computes a one-exact $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-Pareto set $\mathcal{P}$ of size $|\mathcal{P}| \in \mathcal{O}((\frac{M}{\log(1+\varepsilon)})^{d-1})$. The running time is dominated by $\mathcal{O}((\frac{M}{\log(1+\varepsilon)})^{d-1})$ calls to solve $\boldsymbol{\beta}$-RelaxedDualRestrict.*

By Theorem 6, a (randomized) polynomial time oracle for $\boldsymbol{\beta}$-RelaxedDualRestrict is sufficient to compute one-exact $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-Pareto sets in (randomized) polynomial time. Further, we can show that a (randomized) polynomial time oracle for $\boldsymbol{\beta}$-RelaxedDualRestrict is necessary to compute one-exact $(1, \beta_2, \ldots, \beta_d)$-Pareto sets in (randomized) polynomial time.

▶ **Theorem 7.** *Let $I = (X, \boldsymbol{f}) \in \mathcal{I}$ be an instance of a d-objective minimization problem, and $\boldsymbol{\beta} = (\beta_2, \ldots, \beta_d)$ with $\beta_i \colon \mathcal{I} \to \mathbb{Q}_{\geq 1}$ for all $i \in [d]_{\geq 2}$. If a one-exact $(1, \beta_2, \ldots, \beta_d)$-Pareto set can be found in time (randomized) polynomial in $\langle I \rangle$, M, and some (typically constant) function $T(\boldsymbol{\beta})$ then $\boldsymbol{\beta}$-RelaxedDualRestrict can be solved in time (randomized) polynomial in $\langle I \rangle$, M, and $T(\boldsymbol{\beta})$.*

Proof. Assume a one-exact $(1, \beta_2, \ldots, \beta_d)$-Pareto set $\mathcal{P}$ can be found in the required time. To solve $\boldsymbol{\beta}$-RelaxedDualRestrict$(\boldsymbol{b})$, let $\mathcal{P}' := \{x \in \mathcal{P} \mid f_i(x) \leq \beta_i b_i,\ \forall i \in [d]_{\geq 2}\}$. Clearly, $\langle \boldsymbol{b} \rangle$ can be assumed to be polynomial in M. If $\mathcal{P}' = \emptyset$, we return NO; this is correct, since if there was a solution $z \in X$ with $f_i(z) \leq b_i$ for all $i \in [d]_{\geq 2}$, there would be no solution $x \in \mathcal{P}$ that $(1, \beta_2, \ldots, \beta_d)$-approximates z. Otherwise, we return the solution $x \in \mathcal{P}'$ with minimum f_1-value, i.e., $f_1(x) \leq f_1(x')$ for all $x' \in \mathcal{P}'$. Assume $f_1(x) > \mathrm{OPT}_1(\boldsymbol{b})$. Then there would be another solution $z \in X$ with $f_i(z) \leq \beta_i b_i$ for all $i \in [d]_{\geq 2}$ and $f_1(z) < f_1(x)$, but then z is not $(1, \beta_2, \ldots, \beta_d)$-approximated by any $x \in \mathcal{P}$. ◀

Hence, if $\boldsymbol{\beta}$-RelaxedDualRestrict cannot be solved in (randomized) polynomial time, then no $(1, \beta_2, \ldots, \beta_d)$-Pareto set can be computed in (randomized) polynomial time. Thus, up to a factor of $(1+\varepsilon)$, $\boldsymbol{\beta}$-RelaxedDualRestrict is just the appropriate auxiliary problem for computing one-exact $(1, (1+\varepsilon)\beta_2, \ldots, (1+\varepsilon)\beta_d)$-Pareto sets. Note that since there are biobjective problems with exponentially large non-dominated sets, no general framework can guarantee polynomially sized *two-exact* $(1, 1, (1+\varepsilon)\beta_3, \ldots, (1+\varepsilon)\beta_d)$-Pareto sets.

Algorithm 2 A $(1,(1+\varepsilon)\beta)$-approximation for biobjective minimization problems.

Input: instance $(X,\boldsymbol{f})$ of a biobjective minimization problem, $\varepsilon > 0$

$\mathcal{P} \leftarrow$ empty stack, $b \leftarrow 2^M$
loop
  $x \leftarrow$ solve β-RelaxedDualRestrict(b)
  if $x =$ NO **then return** $\mathcal{P}$
  if $|\mathcal{P}| > 0$ and $f_1(\mathcal{P}.\text{top}()) = f_1(x)$ **then** $\mathcal{P}.\text{pop}()$
  $\mathcal{P}.\text{push}(x)$
  $b \leftarrow \frac{f_2(x)}{(1+\varepsilon)\beta}$

3.2 Biobjective Problems

For the special case of $d = 2$ objectives, we can further improve our framework. Since in the biobjective case, the budget vector $\boldsymbol{b} = (b_2)$ and the accuracy vector $\boldsymbol{\beta} = (\beta_2)$ only contain single entries, we can simplify the notation to $b := b_2$ and $\beta := \beta_2$. The full framework is shown in Algorithm 2. Example 8 and Figure 1 give a small demonstration. Contrary to Algorithm 1, which uses a predefined grid of hyperstrips, for biobjective problems we can achieve better guarantees by leveraging adaptively adjusted hyperstrips.

In a loop, we iteratively construct the approximate Pareto set $\mathcal{P}$ as a stack. In each iteration, we want to add one solution to $\mathcal{P}$. Given a current top solution $\bar{x}$ on $\mathcal{P}$, we search for subsequent solutions via β-RelaxedDualRestrict. Thereby, the budget b adaptively grows tighter, depending on the f_2-value of the current top solution $\bar{x}$. Every candidate solution $x \in X$ found for such a budget has a strictly lower f_2-value than the current top solution $\bar{x}$. If x retains the f_1-value of $\bar{x}$, x replaces $\bar{x}$ on top of the stack. Otherwise, x is added on top of $\bar{x}$. The process stops once we cannot find any new solutions.

▶ **Example 8.** Figure 1 shows a small example. For $j \in [6]$, Algorithm 2 discovers solutions $x^{\underline{j}}$ in that order by solving β-RelaxedDualRestrict$(b^{\underline{j}})$ for budget $b^{\underline{j}}$. For each $j \in [6]$, let $\boldsymbol{y}^{\underline{j}} := \boldsymbol{f}(x^{\underline{j}})$. There are no solutions whose points lie within the grey area. All solutions whose points are within the objective space indicated by the orange, blue, and red arrows are $(1,(1+\varepsilon)\beta)$-approximated by $x^{\underline{2}}$, $x^{\underline{5}}$, and $x^{\underline{6}}$, respectively. Solving β-RelaxedDualRestrict$(b^{\underline{7}})$ yields a NO and the algorithm returns $\mathcal{P} = \{x^{\underline{2}}, x^{\underline{5}}, x^{\underline{6}}\}$. ◇

Now, let $\mathcal{P}$ be the set of solutions returned by Algorithm 2. We show that $\mathcal{P}$ is a one-exact $(1,(1+\varepsilon)\beta)$-Pareto set. For Observation 9 and Lemma 10, for $j \in [|\mathcal{P}|]$, let $x^{(j)}$ be the j-th solution on $\mathcal{P}$ (in the order of their addition to $\mathcal{P}$) and let $b^{(j)}$ be the corresponding budget for which solving β-RelaxedDualRestrict$(b^{(j)})$ returned $x^{(j)}$.

▶ **Observation 9.** *For all $j \in [|\mathcal{P}|-1]$, $b^{(j+1)} \leq \frac{b^{(j)}}{1+\varepsilon}$, since $b^{(j+1)} \leq \frac{f_2(x^{(j)})}{(1+\varepsilon)\beta} \leq \frac{b^{(j)}\beta}{(1+\varepsilon)\beta} = \frac{b^{(j)}}{(1+\varepsilon)}$.*

▶ **Lemma 10.** *For every $z \in X$, there is an $x \in \mathcal{P}$ that $(1,(1+\varepsilon)\beta)$-approximates z, i.e., (1) $f_1(x) \leq f_1(z)$ and (2) $f_2(x) \leq (1+\varepsilon)\beta f_2(z)$.*

Proof. Let j be such that $b^{(j)}$ is the minimum budget with $b^{(j)} \geq f_2(z)$. We show that either $x = x^{(j)}$ or $x = x^{(j+1)}$. By the oracle, $f_1(x^{(j)}) \leq \text{OPT}_1(b^{(j)}) \leq f_1(z)$, establishing (1) for $x = x^{(j)}$. Assume (2) does not hold for $x^{(j)}$. In the subsequent iteration, the framework considers a bound $b' = \frac{f_2(x^{(j)})}{(1+\varepsilon)\beta}$, yielding x'. By assumption, $b' > f_2(z)$ and consequently $f_1(z) \geq \text{OPT}_1(b')$. By the former and our choice of j, x' will have been later popped from $\mathcal{P}$ in favor of a solution $x^{(j+1)}$ with $f_1(x^{(j+1)}) = f_1(x') = \text{OPT}_1(b') \leq f_1(z)$ (establishing (1) for $x = x^{(j+1)}$). Since $b^{(j+1)} < f_2(z)$, we have $f_2(x^{(j+1)}) \leq \beta b^{(j+1)} < \beta f_2(z)$. ◀

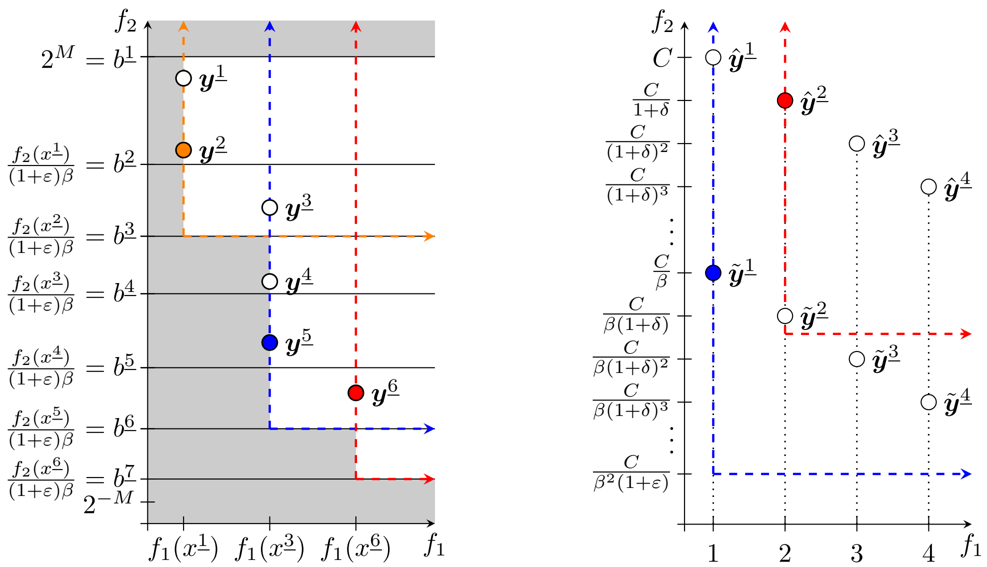

Figure 1 Left: Example 8 for Algorithm 2. Right: Example for Theorem 16 with $k = 4$. Let $\hat{\boldsymbol{y}}^{\underline{j}} := \boldsymbol{f}(\hat{x}^{\underline{j}})$ and $\tilde{\boldsymbol{y}}^{\underline{j}} := \boldsymbol{f}(\tilde{x}^{\underline{j}})$, for all $j \in [4]$. Solutions whose points are within the objective space indicated by the blue and red arrows are $(1, (1+\varepsilon)\beta)$-approximated by $\tilde{x}^{\underline{1}}$ and $\hat{x}^{\underline{2}}$, respectively.

▶ **Lemma 11.** *Algorithm 2 yields a set $\mathcal{P}$ with $|\mathcal{P}| \in \mathcal{O}(\frac{M}{\log(1+\varepsilon)})$ and $|\mathcal{P}| \leq |f_1(X)|$. The total running time of Algorithm 2 is dominated by $\mathcal{O}(\frac{M}{\log(1+\varepsilon)})$ many calls to solve β-RelaxedDualRestrict.*

Proof. We bound the number of times we decrease the budget b until the obtained solutions cover the entire co-domain of f_2 using Observation 9. Recall that $\min_{x \in X} f_2(x) \geq \frac{1}{2^M}$. For Algorithm 2 to return a set $\mathcal{P}$ of size $j \in \mathbb{N}$, we still have to have $\frac{2^M}{(1+\varepsilon)^j} \geq \frac{1}{2^M}$. Thus, $2^{2M} \geq (1+\varepsilon)^j$ and so $j \leq \frac{2M \log 2}{\log(1+\varepsilon)} \in \mathcal{O}(\frac{M}{\log(1+\varepsilon)})$. Further, for every unique value in the image of f_1, $\mathcal{P}$ contains at most one solution. ◀

Hence, choosing, for example $\varepsilon = \log M$ results in a set $\mathcal{P}$ of size $\mathcal{O}(\frac{M}{\log(\log M)})$, which is sublinear in M. Lemmas 10 and 11 establish that Algorithm 2 yields a polynomially sized $(1, (1+\varepsilon)\beta)$-Pareto set in (randomized) polynomial time given that β-RelaxedDualRestrict can be solved in (randomized) polynomial time. Furthermore, we can show that the returned set is at most $\mathcal{O}(\log \beta)$ the size of the minimum-size $(1, (1+\varepsilon)\beta)$-Pareto set $\mathcal{P}^{\min}$.

▶ **Lemma 12.** *For all $x, x' \in \mathcal{P}$ if $f_2(x) \geq f_2(x')$, then $f_2(x) \geq (1+\varepsilon) f_2(x')$.*

Proof. By Observation 9, all considered budgets are at least factor $(1+\varepsilon)\beta$ away from the last solution. The accuracy β leaves a difference of factor at least $(1+\varepsilon)$. ◀

▶ **Lemma 13.** *Let $x' \in \mathcal{P}$. There is no $z \in X$ with $f_1(z) \leq f_1(x')$ and $f_2(z) \leq \frac{f_2(x')}{(1+\varepsilon)\beta}$.*

Proof. Let x be the subsequent answer to β-RelaxedDualRestrict(b) with $b = \frac{f_2(x')}{(1+\varepsilon)\beta}$, at the time when x' is the current top of the stack $\mathcal{P}$ in Algorithm 2. If $x = \text{NO}$, then there is no $z \in X$ with $f_2(z) \leq b$. Otherwise, $x \in X$ is a solution, and in line 5, $f_1(x') \neq f_1(x)$ since otherwise x' would have been removed from $\mathcal{P}$. Hence, $f_1(x') < f_1(x) \leq \text{OPT}_1(b)$. Thus, every $z \in X$ with $f_2(z) \leq b$ has $f_1(z) > f_1(x')$. ◀

▶ **Lemma 14.** *Algorithm 2 yields a $(1,(1+\varepsilon)\beta)$-Pareto set $\mathcal{P}$ with $|\mathcal{P}| \leq (2 \cdot \lceil \log_{1+\varepsilon} \beta \rceil + 2) \cdot |\mathcal{P}^{\min}|$ where $\mathcal{P}^{\min}$ denotes a minimum-size $(1,(1+\varepsilon)\beta)$-Pareto set.*

Proof. Let $k = 2 \cdot \lceil \log_{1+\varepsilon} \beta \rceil + 2$. First, we show that no solution can $(1,(1+\varepsilon)\beta)$-approximate more than k solutions in $\mathcal{P}$. Let $x^{(1)}, x^{(2)}, \ldots, x^{(k)}, x^{(k+1)} \in \mathcal{P}$ be any $k+1$ pairwise different solutions with $f_2(x^{(1)}) \geq f_2(x^{(2)}) \geq \cdots \geq f_2(x^{(k+1)})$. Assume $z \in X$ $(1,(1+\varepsilon)\beta)$-approximates $x^{(j)}$ for all $j \in [k]$; then z does not $(1,(1+\varepsilon)\beta)$-approximate $x^{(k+1)}$. By Lemma 12, $f_2(x^{(j)}) \geq (1+\varepsilon) f_2(x^{(j+1)})$ for all $j \in [k]$. Thus, $f_2(x^{(1)}) \geq (1+\varepsilon)^k \cdot f_2(x^{(k+1)}) = (1+\varepsilon)^{2+2\cdot\lceil \log_{1+\varepsilon} \beta \rceil} \cdot f_2(x^{(k+1)}) \geq ((1+\varepsilon)\beta)^2 \cdot f_2(x^{(k+1)})$. Since z $(1,(1+\varepsilon)\beta)$-approximates $x^{(1)}$, we have $f_1(z) \leq f_1(x^{(1)})$. And thus, by Lemma 13, $f_2(z) > \frac{f_2(x^{(1)})}{(1+\varepsilon)\beta} \geq \frac{((1+\varepsilon)\beta)^2 \cdot f_2(x^{(k+1)})}{(1+\varepsilon)\beta} = (1+\varepsilon)\beta \cdot f_2(x^{(k+1)})$, and z does not $(1,(1+\varepsilon)\beta)$-approximate $x^{(k+1)}$. Now, let $\mathcal{P}^{\min}$ be a minimum-size $(1,(1+\varepsilon)\beta)$-Pareto set. For every $x \in \mathcal{P}$ there must be a $z \in \mathcal{P}^{\min}$ that $(1,(1+\varepsilon)\beta)$-approximates x. Since any $z \in X$ can $(1,(1+\varepsilon)\beta)$-approximate at most k-many $x \in \mathcal{P}$, we get $|\mathcal{P}| \leq k \cdot |\mathcal{P}^{\min}|$. ◀

Combining Lemma 14 and Lemma 4, we can deduce that the size of $\mathcal{P}$ is at most in $\mathcal{O}(\frac{M}{\log((1+\varepsilon)\beta)} \cdot \log \beta)$ which is in $\mathcal{O}(M)$ for any constant $\varepsilon > 0$. Overall, we have:

▶ **Theorem 15.** *For any $\varepsilon > 0$, and any instance $(X, \boldsymbol{f})$ of a biobjective minimization problem, Algorithm 2 yields a one-exact $(1,(1+\varepsilon)\beta)$-Pareto set $\mathcal{P}$ of size $|\mathcal{P}| \in \mathcal{O}(\frac{M}{\log(1+\varepsilon)})$, $|\mathcal{P}| \leq |f_1(X)|$, and $|\mathcal{P}|$ is at most $(2 \cdot \lceil \log_{1+\varepsilon} \beta \rceil + 2)$ times the size of the smallest one-exact $(1,(1+\varepsilon)\beta)$-Pareto set. The running time is dominated by $\mathcal{O}(\frac{M}{\log(1+\varepsilon)})$ calls to solve β-RelaxedDualRestrict.*

We show that any framework that yields $(1,(1+\varepsilon)\beta)$-Pareto sets for biobjective minimization problems that is based on RelaxedDualRestrict must solve instances of β'-RelaxedDualRestrict with $\beta' < \beta$ in order to guarantee $|\mathcal{P}| < (\lfloor \log_{1+\varepsilon}(\beta) \rfloor + 1) \cdot |\mathcal{P}^{\min}|$.

▶ **Theorem 16.** *For any $\delta > \varepsilon > 0$, there is no algorithm that, for every biobjective minimization problem instance, computes a $(1,(1+\varepsilon)\beta)$-Pareto set $\mathcal{P}$ with $|\mathcal{P}| < \lfloor (\log_{1+\delta}((1+\varepsilon)\beta) \rfloor \cdot |\mathcal{P}^{\min}|$, and generates solutions only by solving β-RelaxedDualRestrict. For $\delta \to \varepsilon$, this approaches $|\mathcal{P}| < (\lfloor \log_{1+\varepsilon}(\beta) \rfloor + 1) \cdot |\mathcal{P}^{\min}|$.*

Proof. Consider an instance $I = (X = \hat{X} \cup \tilde{X}, \boldsymbol{f})$ with $|\hat{X}| = |\tilde{X}| = \lfloor (\log_{1+\delta}((1+\varepsilon)\beta) \rfloor = k \in \mathbb{N}$ and $\hat{X} = \{\hat{x}^{\underline{1}}, \ldots, \hat{x}^{\underline{k}}\}$, $\tilde{X} = \{\tilde{x}^{\underline{1}}, \ldots, \tilde{x}^{\underline{k}}\}$. For all $j \in [k]$, $f_1(\hat{x}^{\underline{j}}) = f_1(\tilde{x}^{\underline{j}}) = j$, $f_2(\hat{x}^{\underline{j}}) = \frac{C}{(1+\delta)^{j-1}}$ and $f_2(\tilde{x}^{\underline{j}}) = \frac{C}{\beta \cdot (1+\delta)^{j-1}}$ for some $C > \beta^2(1+\varepsilon)$. This is illustrated in Figure 1. Note that $\tilde{x}^{\underline{1}}$ has minimum possible value in f_1 and $f_2(\tilde{x}^{\underline{k}}) = \frac{C}{\beta \cdot (1+\delta)^{k-1}} \geq \frac{C}{\beta \cdot \beta(1+\varepsilon)} = \frac{f_2(\tilde{x}^{\underline{1}})}{\beta(1+\varepsilon)}$. Thus, $\tilde{x}^{\underline{1}}$ $(1,(1+\varepsilon)\beta)$-approximates all $x \in X$ and $\mathcal{P}^{\min} = \{\tilde{x}^{\underline{1}}\}$ is the minimum-size $(1,(1+\varepsilon)\beta)$-Pareto set.

We show that algorithms that generate solutions only via β-RelaxedDualRestrict cannot guarantee a $(1,(1+\varepsilon)\beta)$-Pareto set of size $|\mathcal{P}| < |\hat{X}| = k$. For different budgets b, consider the results of β-RelaxedDualRestrict(b). For all $b > f_2(\tilde{x}^{\underline{1}})$, we get $\hat{x}^{\underline{1}}$. For all $b < f_2(\tilde{x}^{\underline{k}})$, we get NO as there is no feasible solution. For any $f_2(\tilde{x}^{\underline{1}}) \leq b \leq f_2(\tilde{x}^{\underline{k}})$, let $\tilde{x}^{\underline{j}}$ be the solution with maximum f_2-value and $f_2(\tilde{x}^{\underline{j}}) \leq b$. Then $\hat{x}^{\underline{j}}$ is a feasible solution to β-RelaxedDualRestrict(b) as $f_1(\hat{x}^{\underline{j}}) \leq \mathrm{OPT}_1(b)$ and $f_2(\hat{x}^{\underline{j}}) \leq \beta b$. For all budgets b that do not incite a NO, we may get an $\hat{x}^{\underline{j}}$. Hence, only using β-RelaxedDualRestrict we may get $\mathcal{P} = \hat{X}$ with $|\hat{X}| = k = \lfloor (\log_{1+\delta}((1+\varepsilon)\beta) \rfloor$, which approaches $\lfloor \log_{1+\varepsilon}(\beta) \rfloor + 1$, for $\delta \to \epsilon$. Note that, for any $j \in [k]$, $\hat{X} \setminus \{\hat{x}^{\underline{j}}\}$ no longer is a $(1,(1+\varepsilon)\beta)$-Pareto set. ◀

We can show that the relative size guarantee of Algorithm 2 is asymptotically tight by putting the bounds of Lemma 14 in relation to those of Theorem 16. Let $\lambda := \lceil \log_{1+\varepsilon}(\beta) \rceil$, then $\lfloor \log_{1+\varepsilon}(\beta) \rfloor \geq \lambda - 1$ and $\frac{2 \cdot \lceil \log_{1+\varepsilon}(\beta) \rceil + 2}{\lfloor \log_{1+\varepsilon}(\beta) \rfloor + 1} \leq \frac{2\lambda+2}{\lambda-1+1} = 2 + \frac{2}{\lambda} \leq 4$, for all $\lambda \in \mathbb{N}$.

▶ **Corollary 17.** *The relative size guarantee w.r.t. the minimum-size one-exact* $(1, (1+\varepsilon)\beta)$*-Pareto set obtained by Algorithm 2 is tight up to a factor of* 4*. For non-constant (increasing)* β*, this ratio approaches* 2*.*

4 Finding Oracles for β-RelaxedDualRestrict

Finding a (randomized) polynomial-time oracle for $\boldsymbol{\beta}$-RelaxedDualRestrict of a given MOO problem is often non-trivial. We show how to adapt existing approximation algorithms for single-objective problems into suitable $\boldsymbol{\beta}$-RelaxedDualRestrict oracles. In Section 5, we use these techniques to derive new best approximations for various MOO problems.

$\boldsymbol{\beta}$-RelaxedRestrict. Similar to how $\boldsymbol{\beta}$-RelaxedDualRestrict is a relaxed version of $\text{DualRestrict}_{1+\delta}$, we can analogously define a relaxed version of $\text{Restrict}_{1+\delta}$ [24].

▶ **Definition 18** (β-RelaxedRestrict)**.** *Let* $\mathcal{I}$ *be the set of instances of a* d*-objective minimization problem. Let* $\beta \colon \mathcal{I} \to \mathbb{Q}_{\geq 1}$ *be polynomial-time computable. Given an instance* $I = (X, \boldsymbol{f}) \in \mathcal{I}$ *and a budget vector* $\boldsymbol{c} = (c_1, \ldots, c_{d-1}) \in \mathbb{Q}^{d-1}_{>0}$*, the auxiliary problem* β*-RelaxedRestrict asks for an* $x \in X$ *with* $f_i(x) \leq c_i$ *for all* $i \in [d-1]$ *and* $f_d(x) \leq \beta(I)\,OPT_d(\boldsymbol{c})$*. If no such solution* x *exists, the answer shall be* $x = NO$*.*

Again, we do not explicitly state I as part of the input to β and β-RelaxedRestrict in the following. For biobjective problems, β-RelaxedRestrict corresponds to a β-approximation of the single-objective problem minimizing f_2 while constraining f_1 via $c \coloneqq c_1$. For any $\delta > 0$, $\text{Restrict}_{1+\delta}$ and $\text{DualRestrict}_{1+\delta}$ are known to be polynomially equivalent via binary search [24] for biobjective problems. Analogously, we have Lemma 19.

▶ **Lemma 19.** *β-RelaxedRestrict and β-RelaxedDualRestrict are polynomially equivalent for any biobjective minimization problem.*

Proof. Let $(X, \boldsymbol{f})$ be an instance to a biobjective minimization problem. Assume we want to solve β-RelaxedDualRestrict(b) using a β-RelaxedRestrict oracle. If solving β-RelaxedRestrict(2^M) returns NO or a solution $x \in X$ with $f_2(x) > \beta b$, we return NO, since there is no $x' \in X$ with $f_2(x') \leq b$. Otherwise, we perform a binary search for the smallest budget $c' \in [2^{-M}, 2^M]$ such that solving β-RelaxedRestrict(c') returns a solution $x \in X$ with $f_1(x) \leq c'$ and $f_2(x) \leq \beta b$. We return this x, since $c' \leq \text{OPT}_1(b)$. The binary search only requires a granularity of 2^{-2M}, since that is the minimum difference between two objective function values. Since the number of solved β-RelaxedRestrict instances is in $\Theta(M)$, the running time of the constructed β-RelaxedDualRestrict oracle is polynomial in the running time of the given β-RelaxedRestrict oracle.

On the other hand, analogously, assume we want to solve β-RelaxedRestrict(c) using a β-RelaxedDualRestrict oracle. If solving β-RelaxedDualRestrict(2^M) returns NO or a solution $x \in X$ with $f_1(x) > c$, we return NO. Otherwise, we perform a binary search for the smallest $b' \in [2^{-M}, 2^M]$ such that solving β-RelaxedDualRestrict(b') returns a solution $x \in X$ with $f_1(x) \leq c$ and $f_2(x) \leq \beta b'$. The analyses of correctness and running time are analogous to those of the first direction. ◀

Consequently, in order to apply our framework for biobjective problems, it suffices to have an oracle for β-RelaxedRestrict that has (randomized) polynomial running time. Further, if no (randomized) polynomial-time oracle for β-RelaxedRestrict exists, then there is no (randomized) polynomial-time oracle for β-RelaxedDualRestrict either.

Re-Randomization. Oracles for $\boldsymbol{\beta}$-RelaxedDualRestrict are allowed to have randomized running time guarantees, but must always answer correctly. For single-objective problems, any deterministic-time algorithm that is randomized in either feasibility or approximation ratio can be re-randomized by repetition, yielding a randomized-time algorithm that guarantees both feasibility and the desired ratio. When stochastic guarantees apply to both, repetition alone may not suffice – an algorithm might always return either a superoptimal infeasible solution or a poor feasible one. Nevertheless, re-randomization remains viable even in the case of MOO problems, a scenario for which this seemingly has not been considered before. We may encode (parts of) feasibility as an additional objective function. Consistent with the above, we state the following in the context of multiobjective minimization problems; adaptations to maximization or mixed optimization are trivial. Let us first define the specific algorithms we aim to re-randomize.

▶ **Definition 20** (Multi-Randomized Algorithm). *For a fixed $d \in \mathbb{N}_{\geq 1}$, let $\mathcal{I}$ be the set of instances of a d-objective minimization problem. Let $q\colon \mathcal{I} \to \mathbb{Q}_{>0}$ and $\sigma\colon \mathcal{I} \to (0,1)$. We say an algorithm $\mathcal{A}$ is* multi-randomized *(w.r.t. q and σ) if, for every $I = (X, \boldsymbol{f}) \in \mathcal{I}$, it returns a solution $x \in X$ with $\mathbb{P}[f_1(x) \leq q(I)] \geq \sigma(I)$ and $\mathbb{E}[f_i(x)] > 0$ is finite for each $i \in [d]_{\geq 2}$.*

We estimate the probability that a multi-randomized algorithm $\mathcal{A}$ returns a solution $x \in X$ that simultaneously satisfies $\mathcal{A}$'s guarantee w.r.t. f_1, i.e., $f_1(x) \leq q(I)$, and is close to $\mathcal{A}$'s expected solution values in all other objectives. For the latter, we say a solution $x \in X$ is *φ-good* if $f_i(x) \leq \varphi \cdot \mathbb{E}[f_i(x)]$ for all $i \in [d]_{\geq 2}$ and some $\varphi \geq 1$.

▶ **Theorem 21.** *For $0 < \lambda < 1$, a multi-randomized algorithm $\mathcal{A}$ has probability at least $\lambda \cdot \sigma(I)$ to return a φ-good solution $x \in X$ with $f_1(x) \leq q(I)$ and $\varphi = \frac{d-1}{(1-\lambda)\sigma(I)}$.*

Proof. First, we estimate the probability that the algorithm's solution x satisfies $f_1(x) \leq q(I)$ under the condition that x is φ-good. By Markov's inequality for each individual $i \in [d]_{\geq 2}$ and union bound summing over all these i, we have $\mathbb{P}[x \text{ is not } \varphi\text{-good}] \leq \frac{d-1}{\varphi}$ and thus $\mathbb{P}[x \text{ is } \varphi\text{-good}] \geq \frac{\varphi-d+1}{\varphi}$. Let Z denote the property of x being φ-good. We get $\mathbb{P}[f_1(x) \leq q(I) \mid Z] = 1 - \mathbb{P}[f_1(x) > q(I) \mid Z] = 1 - \frac{\mathbb{P}[f_1(x) > q(I) \wedge Z]}{\mathbb{P}[Z]} \geq 1 - \frac{\mathbb{P}[f_1(x) > q(I)]}{\mathbb{P}[Z]} \geq 1 - \frac{1-\sigma(I)}{\frac{\varphi-d+1}{\varphi}} = 1 - \frac{\varphi(1-\sigma(I))}{\varphi-d+1}$. We estimate the probability that the solution is simultaneously φ-good and satisfies $f_1(x) \leq q(I)$. We have $\mathbb{P}[Z \cap f_1(x) \leq q(I)] = \mathbb{P}[Z] \cdot \mathbb{P}[f_1(x) \leq q(I) \mid Z] \geq (\frac{\varphi-d+1}{\varphi}) \cdot (1 - \frac{\varphi(1-\sigma(I))}{\varphi-d+1}) = \frac{1-d}{\varphi} + \sigma(I)$. For $\varphi = \frac{d-1}{(1-\lambda)\sigma(I)}$ we have $\frac{1-d}{\varphi} + \sigma(I) \geq \lambda \cdot \sigma(I)$. ◀

We can derive Corollary 22 using the geometric probability distribution. See Section 5 for an application in the context of β-RelaxedRestrict for Multiobjective Spanners.

▶ **Corollary 22.** *For $0 < \lambda < 1$, after expected $\frac{1}{\lambda \cdot \sigma(I)}$ many independent attempts, $\mathcal{A}$ yields a φ-good solution $x \in X$ with $f_1(x) \leq q(I)$ for $\varphi = \frac{d-1}{(1-\lambda)\sigma(I)}$.*

5 Applications

Lastly, we showcase how to achieve new or first approximation guarantees for several MOO problems using our frameworks. Table 1 gives an overview. First, we show explicitly how our methods can be applied to approximate the Multiobjective Spanner problem. Afterwards, we briefly discuss a selection of further established MOO problems where our methods achieve new best approximation guarantees, some of which are (asymptotically) best-possible.

Multiobjective Spanners. We demonstrate our framework in the context of the *Multiobjective Spanner* problem [16]. Up to now, there are no known polynomial-time algorithms that compute any approximate Pareto sets for this problem. Given a connected graph $G = (V, E)$ with n nodes, weights $w\colon E \to \mathbb{Q}_{>0}$, and lengths $\ell\colon E \to \mathbb{Q}_{>0}$. A *spanner* is a connected subgraph $H = (V, E_H)$ with $E_H \subseteq E$ that should minimize both the *stretch* $f_1(H) = s(H) = \max_{u,v\in V} \frac{\ell_H[u,v]}{\ell_G[u,v]}$ and weight $f_2(H) = w(H) = \sum_{e\in E_H} w(e)$. Thereby, the distance $\ell_{G'}[u, v]$ is the length of a shortest path from u to v in G', subject to lengths ℓ. Spanners find applications in, e.g., communication networks [59, 60], approximate distance oracles [58, 62], graph drawing [4, 65], and access control hierarchies [12, 43]. Even on degree-3 outerplaner graphs, the Multiobjective Spanner problem is both intractable and **BUCO**-hard [16]. Unless $\mathbf{P} = \mathbf{NP}$, neither the non-dominated set $\mathcal{Y}^*$ nor its subset of *extreme points* can be computed in output-polynomial time, even if $w(e) = \ell(e) = 1$ for all $e \in E$ [16].

For the Multiobjective Spanner problem, β-RelaxedRestrict(c) asks to either answer NO if there is no spanner with stretch at most c or return a spanner H with stretch $s(H) \leq c$ and weight $w(H) \leq \beta\mathrm{OPT}_2(c)$. Trivially, we need to answer NO if and only if $c < 1$. For $c \geq 1$, the problem coincides with approximating the single-objective *Minimum α-Spanner* problem up to a factor of β with $\alpha = c$: Given a stretch $\alpha \geq 1$, find a spanner H with $s(H) \leq \alpha$ of minimum total weight $w(H)$. Even for weights and lengths polynomially bounded in n, the Minimum α-Spanner problem is **NP**-hard [35, 59] and allows no $2^{\log^{1-\varepsilon} n}$-approximation, for any $1 < \alpha \in \mathcal{O}(n^{1-\varepsilon'})$ and $\varepsilon, \varepsilon' \in (0, 1)$ [31], unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{\mathrm{polylog}\, n})$. Thus, Restrict$_{1+\delta}$, and by extension DualRestrict$_{1+\delta}$, of the Multiobjective Spanner problem cannot be solved in polynomial time under this assumption, for an arbitrary constant $\delta > 0$. Hence, the only other framework for one-exact approximate Pareto sets [39] is inapplicable. Still, there are many approximations for the Minimum α-Spanner problem. Most of them make simplifying assumptions on weights and lengths [1, 9, 19, 25–27, 50, 51]. Several spanner approximations have been implemented and evaluated in practice [14, 15, 18]. Existing strong approximations for the general Minimum α-Spanner problem [13, 17, 36] require polynomially-bounded integer lengths, have expected approximation ratios, and only yield feasible solutions with high probability. The best known (expected) ratio is achieved by [36].

▶ **Algorithm 3** (Minimum α-Spanner approximation [36])**.** *For the Minimum α-Spanner problem with polynomially-bounded integer lengths, the algorithm of [36] yields a spanner H with $\mathbb{P}[s(H) \leq \alpha] \geq \frac{n-1}{n}$ and expected weight $\mathbb{E}[w(H)] \leq \bar{\beta}\, OPT_2(\alpha)$ with $\bar{\beta} \in \widetilde{\mathcal{O}}(n^{4/5+\mu})$ in polynomial time in n, for any constant $\mu > 0$.*

The $\widetilde{\mathcal{O}}$-notation hides polylogarithmic factors. Using our methods from Section 4, we can adapt Algorithm 3 to obtain an oracle for β-RelaxedDualRestrict with expected polynomial running time for the Multiobjective Spanner problem with polynomially-bounded integer lengths. We could solve $\bar{\beta}$-RelaxedRestrict(c) with randomized solution quality as follows: If $c < 1$, we answer NO. Otherwise, we return the spanner yielded by Algorithm 3. We can re-randomize this by applying Corollary 22 for $\lambda = \frac{1}{2}$. This yields a suitable β-RelaxedRestrict oracle that has expected polynomial running time and deterministic solution quality without degrading the asymptotic guarantee of Algorithm 3: since w.l.o.g. $\frac{n-1}{n} > \frac{1}{2}$ we have $\beta = \varphi\bar{\beta} = \frac{1}{(1-\lambda)\sigma(I)}\bar{\beta} = \frac{1}{(1-\frac{1}{2})\frac{n-1}{n}}\bar{\beta} \leq 4\bar{\beta} \in \mathcal{O}(\bar{\beta})$.

Combining this with Lemma 19, we obtain a β-RelaxedDualRestrict oracle for the Multiobjective Spanner problem with polynomially-bounded integer lengths that has expected polynomial running time. Using this as a subroutine in Algorithm 2 yields Corollary 23.

Table 1 Some guarantees achieved using Theorem 15. To our knowledge, the only previous approximation result for these problems are $(3,3)$-Pareto sets for both Facility Location variants. 'Oracle' solves β-RelaxedRestrict. 'Hard.' gives best known hardness of approximation results for β-RelaxedRestrict. Our approximation ratios marked $*$ are best-possible up to a factor of $1+\gamma$, for any constant $\gamma > 0$, by Theorem 7 under established complexity assumptions. ⚄ marks expected polynomial running time. 'PC' stands for Prize-Collecting. Let $\varepsilon > 0$ and fixed $\mu > 0$.

Biobjective Problem	Objectives f_1, f_2	β	Oracle	Hard.	Our approximation ratio
Spanner [16]	min. stretch, min. weight	$\widetilde{\mathcal{O}}(n^{4/5+\mu})$	[36, Thm. 2]	[31]	⚄ $(1, (1+\varepsilon)\cdot\widetilde{\mathcal{O}}(n^{4/5+\mu}))$
Quasi-Clique [29]	max. size, max. density	$\Omega(\frac{1}{n^{1/4+\mu}})$	[11, Thm. 1.2]	[48]	$(1, (1-\varepsilon)\cdot\Omega(\frac{1}{n^{1/4+\mu}}))$
Vehicle Routing [20]	min. makespan, min. fleet size	$\mathcal{O}(\log n)$	[54, Sec. 1.4]	[21]	$(1, (1+\varepsilon)\cdot \mathcal{O}(\log n))$
Facility Location [5]	min. radius, min. opening+travel cost	$\ln n + 1$	[66, Thm. 1]	[66]	$*$ $(1, (1+\varepsilon)\cdot (\ln n + 1))$
Facility Location [5]	min. opening cost, min. radius	3	[41, Thm. 5]	[41]	$*$ $(1, (1+\varepsilon)\cdot 3)$
PC Set Cover [53]	min. cost, max. profit	$1-\frac{1}{e}$	[49, Thm. 4]	[49]	$*$ $(1, (1-\varepsilon)\cdot (1-\frac{1}{e}))$
PC TSP [46]	min. makespan, max. profit	$\frac{1}{2}$	[56, Thm. 4]	—	$(1, (1-\varepsilon)\cdot \frac{1}{2})$
PC Steiner Tree [52]	min. cost, max. profit	$\frac{1}{2}$	[56, Thm. 5]	—	$(1, (1-\varepsilon)\cdot \frac{1}{2})$
Firefighter [42]	max. survivors, min. fleet size	$\ln n$	[3, Thm. 5]	[3]	$*$ $(1, (1+\varepsilon)\cdot \ln n)$

▶ **Corollary 23.** *For the Multiobjective Spanner problem with polynomially-bounded integer edge lengths, for any $\varepsilon, \mu > 0$, Algorithm 2 yields a $(1, (1+\varepsilon)\widetilde{\mathcal{O}}(n^{4/5+\mu}))$-Pareto set. Its size is $|\mathcal{P}| \in \mathcal{O}(\frac{M}{\log(1+\varepsilon)})$, $|\mathcal{P}| \leq |f_1(X)|$, and it is at most $(2 \cdot \lceil \log_{1+\varepsilon} \widetilde{\mathcal{O}}(n^{4/5+\mu}) \rceil + 2) \in \mathcal{O}(\log n)$ times the size of the smallest $(1, (1+\varepsilon)\widetilde{\mathcal{O}}(n^{4/5+\mu}))$-Pareto set. The total running time is expected polynomial in n.*

Further Applications. We briefly discuss a selection of further established MOO problems, where our methods find application. Table 1 gives an overview of our results. See the respective references for detailed problem definitions (or Appendix A for a synopsis). Analogous to the Multiobjective Spanner problem, the key insight is that for each of these problems, the β-RelaxedRestrict problem essentially coincides with approximating a well-studied single-objective problem with ratio β. Answering NO correctly is typically trivial. By Lemma 19, this gives us an oracle for the β-RelaxedDualRestrict problem. We remark that some of our approximation guarantees are best-possible up to a factor of $1+\gamma$, for any constant $\gamma > 0$, by Theorem 7 under standard complexity assumptions ($1-\gamma$ if f_2 is maximized, respectively).

For biobjective problems where the image of function f_1, only has polynomially many unique values, one can *brute force* a polynomially sized one-exact $(1,\beta)$-Pareto set $\mathcal{P}$ in polynomial time by solving β-RelaxedRestrict(c) in polynomial time for each $c \in f_1(X)$ and adding resulting solutions to $\mathcal{P}$. However, in contrast to our framework, this achieves no relative size guarantees w.r.t. the smallest approximate Pareto set of that quality. In [5, Thm. 45], they use a brute force strategy to obtain $(3,3)$-Pareto sets for both variations of the Biobjective Facility Location problem shown in Table 1. For the variation with and without travel costs, we obtain one-exact $(1, (1+\varepsilon)(\ln n + 1))$- and $(1, (1+\varepsilon)3)$-Pareto sets, respectively. Both guarantees are (asymptotically) best-possible. Interestingly, for the variation with travel cost, our result highlights that being one-exact w.r.t. one objective may come at the cost of a worse accuracy in the remaining objectives. To our knowledge, none of the other problems shown in Table 1 have known approximation algorithms. Furthermore, none of the previously existing approximation frameworks are applicable as either no polynomial-time oracle for the corresponding auxiliary problem is known or no such oracle can exist under established complexity assumptions. Technically, in the biobjective case, the only other known framework for one-exact approximate Pareto sets [39] could be used with a non-PTAS oracle; this use is never discussed in the paper and certain guarantees may break, which we are not aware of. Anyhow, even if the algorithm stays correct, its approximation guarantees would drastically inflate by a power of 3, i.e., if accuracy β is the best-possible ratio to solve DualRestrict$_\beta$, their framework would only yield $(1,(1+\varepsilon)\beta^3)$-Pareto sets.

References

1 Ingo Althöfer, Gautam Das, David Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993. doi:10.1007/BF02189308.

2 Eric Angel, Evripidis Bampis, and Alexander Kononov. On the approximate tradeoff for bicriteria batching and parallel machine scheduling problems. *Theoretical Computer Science*, 306(1-3):319–338, 2003. doi:10.1016/S0304-3975(03)00288-3.

3 Elliot Anshelevich, Deeparnab Chakrabarty, Ameya Hate, and Chaitanya Swamy. Approximability of the firefighter problem: Computing cuts over time. *Algorithmica*, 62(1):520–536, 2012. doi:10.1007/s00453-010-9469-y.

4 Daniel Archambault, Giuseppe Liotta, Martin Nöllenburg, Tommaso Piselli, Alessandra Tappini, and Markus Wallinger. Bundling-aware graph drawing. In *Proc. GD 2024*, volume 320, pages 15:1–15:19, Dagstuhl, 2024. LZI Dagstuhl. doi:10.4230/LIPIcs.GD.2024.15.

5 Anna Arutyunova, Jan Eube, Heiko Röglin, Melanie Schmidt, Sarah Sturm, and Julian Wargalla. Approximately pareto-optimal solutions for bi-objective k-clustering. *Advances in Neural Information Processing Systems*, 37:109517–109581, 2024. doi:10.52202/079017-3477.

6 Cristina Bazgan, Laurent Gourvès, and Jérôme Monnot. Approximation with a fixed number of solutions of some multiobjective maximization problems. *Journal of Discrete Algorithms*, 22:19–29, 2013. doi:10.1016/j.jda.2013.06.006.

7 Cristina Bazgan, Florian Jamain, and Daniel Vanderpooten. Approximate pareto sets of minimal size for multi-objective optimization problems. *Operations Research Letters*, 43(1):1–6, 2015. doi:10.1016/j.orl.2014.10.003.

8 Cristina Bazgan, Stefan Ruzika, Clemens Thielen, and Daniel Vanderpooten. The power of the weighted sum scalarization for approximating multiobjective optimization problems. *Theory of Computing Systems*, 66(1):395–415, 2022. doi:10.1007/s00224-021-10066-5.

9 Piotr Berman, Arnab Bhattacharyya, Konstantin Makarychev, Sofya Raskhodnikova, and Grigory Yaroslavtsev. Approximation algorithms for spanner problems and directed steiner forest. *Information and Computation*, 222:93–107, 2013. doi:10.1016/j.ic.2012.10.007.

10 Jean-François Bérubé, Michel Gendreau, and Jean-Yves Potvin. An exact ε-constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits. *European journal of operational research*, 194(1):39–50, 2009. doi:10.1016/j.ejor.2007.12.014.

11 Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. Detecting high log-densities: an $o(n^{1/4})$ approximation for densest k-subgraph. In *Proc. STOC 2010*, pages 201–210, New York, 2010. ACM. doi:10.1145/1806689.1806719.

12 Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Transitive-closure spanners. *SIAM Journal on Computing*, 41(6):1380–1425, 2012. doi:10.1137/110826655.

13 Fritz Bökler, Markus Chimani, and Henning Jasper. Simple approximations for general spanner problems. In *Proc. Approximation and Online Algorithms*, volume 16077 of *LNCS*, pages 48–63, Cham, 2025. Springer Nature Switzerland. doi:10.1007/978-3-032-06706-7_4.

14 Fritz Bökler, Markus Chimani, and Henning Jasper. General Multiplicative Spanners in Practice. In *Proc. SEA 2026*, volume 371 of *LIPIcs*, pages 8:1–8:21. LZI Dagstuhl, 2026. doi:10.4230/LIPIcs.SEA.2026.8.

15 Fritz Bökler, Markus Chimani, Henning Jasper, and Mirko H. Wagner. Exact minimum weight spanners via column generation. In *Proc. Algorithms–ESA 2024*, volume 308 of *LIPIcs*, pages 30:1–30:17. LZI Dagstuhl, 2024. doi:10.4230/LIPIcs.ESA.2024.30.

16 Fritz Bökler and Henning Jasper. Complexity of the multiobjective minimum weight minimum stretch spanner problem. *Mathematical Methods of Operations Research*, 100(1):65–83, 2024. doi:10.1007/s00186-024-00850-7.

17 Markus Chimani and Joachim Spoerhase. Network design problems with bounded distances via shallow-light steiner trees. In *Proc. STACS 2015*, volume 30 of *LIPIcs*, pages 238–248, Dagstuhl, 2015. LZI Dagstuhl. doi:10.4230/LIPIcs.STACS.2015.238.

18 Markus Chimani and Finn Stutzenstein. Spanner approximations in practice. In *Proc. Algorithms–ESA 2022*, volume 244 of *LIPIcs*, pages 37:1–37:15. LZI Dagstuhl, 2022. doi:10.4230/LIPIcs.ESA.2022.37.

19 Eden Chlamtáč, Michael Dinitz, Guy Kortsarz, and Bundit Laekhanukit. Approximating spanners and directed steiner forest: Upper and lower bounds. *ACM Transactions on Algorithms*, 16(3), 2020. doi:10.1145/3381451.

20 Angel Corberán, Elena Fernández, Manuel Laguna, and Rafael Marti. Heuristic solutions to the problem of routing school buses with multiple objectives. *Journal of the Operational Research Society*, 53(4):427–435, 2002. doi:10.1057/palgrave.jors.2601324.

21 Haipeng Dai, Xiaobing Wu, Lijie Xu, Guihai Chen, and Shan Lin. Using minimum mobile chargers to keep large-scale wireless rechargeable sensor networks running forever. In *Proc. ICCCN 2013*, pages 1–7, New York, 2013. IEEE. doi:10.1109/ICCCN.2013.6614207.

22 Constantinos Daskalakis, Ilias Diakonikolas, and Mihalis Yannakakis. How good is the chord algorithm? *SIAM Journal on Computing*, 45(3):811–858, 2016. doi:10.1137/13093875X.

23 Erik D. Demaine, Timothy D. Goodrich, Kyle Kloster, Brian Lavallee, Quanquan C. Liu, Blair D. Sullivan, Ali Vakilian, and Andrew van der Poel. Structural Rounding: Approximation Algorithms for Graphs Near an Algorithmically Tractable Class. In *Proc. Algorithms–ESA 2019*, volume 144 of *LIPIcs*, pages 37:1–37:15. LZI Dagstuhl, 2019. doi:10.4230/LIPIcs.ESA.2019.37.

24 Ilias Diakonikolas and Mihalis Yannakakis. Small approximate pareto sets for biobjective shortest paths and other problems. *SIAM Journal on Computing*, 39(4):1340–1371, 2010. doi:10.1137/080724514.

25 Michael Dinitz and Robert Krauthgamer. Directed spanners via flow-based linear programs. In *Proc. STOC 2011*, pages 323–332, New York, 2011. ACM. doi:10.1145/1993636.1993680.

26 Michael Dinitz and Zeyu Zhang. Approximating low-stretch spanners. In *Proc. ACM-SIAM SODA 2016*, pages 821–840. SIAM, 2016. doi:10.1137/1.9781611974331.ch59.

27 Yevgeniy Dodis and Sanjeev Khanna. Design networks with bounded pairwise distance. In *Proc. STOC 1999*, pages 750–759, New York, 1999. ACM. doi:10.1145/301250.301447.

28 Daniela dos Santos, Kathrin Klamroth, Pedro Martins, and Luís Paquete. A path-relinking-based heuristic for the multiobjective subgraph problem. In *Proc. GECCO 2025*, pages 304–312, New York, 2025. ACM. doi:10.1145/3712256.372638.

29 Daniela Scherer dos Santos, Kathrin Klamroth, Pedro Martins, and Luís Paquete. Solving the multiobjective quasi-clique problem. *European Journal of Operational Research*, 323(2):409–424, 2025. doi:10.1016/j.ejor.2024.12.018.

30 Matthias Ehrgott. *Multicriteria optimization.* Springer, Berlin, 2005. doi:10.1007/3-540-27659-9.

31 Michael Elkin and David Peleg. The hardness of approximating spanner problems. *Theory of Computing Systems*, 41(4):691–729, 2007. doi:10.1007/s00224-006-1266-2.

32 Mohammad Saied Fallah Niasar. *Metaheuristics for the school bus routing problem.* PhD thesis, University of Antwerp, 2024. doi:10.63028/10067/2043980151162165141.

33 Zachary Friggstad and Chaitanya Swamy. Approximation algorithms for regret-bounded vehicle routing and applications to distance-constrained vehicle routing. In *Proc. STOC 2014*, pages 744–753, New York, 2014. ACM. doi:10.1145/2591796.2591840.

34 Christian Glaßer, Christian Reitwießner, Heinz Schmitz, and Maximilian Witek. Approximability and hardness in multi-objective optimization. In *Proc. Programs, Proofs, Processes*, volume 6158 of *LNCS*, pages 180–189, Berlin, Heidelberg, 2010. Springer. doi:10.1007/978-3-642-13962-8_20.

35 Renzo Gómez, Flávio Keidi Miyazawa, and Yoshiko Wakabayashi. Improved NP-hardness results for the minimum t-spanner problem on bounded-degree graphs. *Theoretical Computer Science*, 947:113691, 2023. doi:10.1016/j.tcs.2023.113691.

36 Elena Grigorescu, Nithish Kumar, and Young-San Lin. Approximation algorithms for directed weighted spanners. In *Proc. APPROX/RANDOM 2023*, volume 275 of *LIPIcs*, pages 8:1–8:23, Dagstuhl, 2023. LZI Dagstuhl. doi:10.4230/LIPIcs.APPROX/RANDOM.2023.8.

37 Pascal Halffmann, Stefan Ruzika, Clemens Thielen, and David Willems. A general approximation method for bicriteria minimization problems. *Theoretical Computer Science*, 695:1–15, 2017. doi:10.1016/j.tcs.2017.07.003.

38 Stephan Helfrich, Stefan Ruzika, and Clemens Thielen. Efficiently constructing convex approximation sets in multiobjective optimization problems. *INFORMS Journal on Computing*, 37(5), 2024. doi:10.1287/ijoc.2023.0220.

39 Arne Herzel, Cristina Bazgan, Stefan Ruzika, Clemens Thielen, and Daniel Vanderpooten. One-exact approximate pareto sets. *Journal of Global Optimization*, 80(1):87–115, 2021. doi:10.1007/s10898-020-00951-7.

40 Arne Herzel, Stefan Ruzika, and Clemens Thielen. Approximation methods for multiobjective optimization problems: A survey. *INFORMS Journal on Computing*, 33(4):1284–1299, 2021. doi:10.1287/ijoc.2020.1028.

41 Dorit S. Hochbaum and David B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. *Journal of the ACM*, 33(3):533–550, 1986. doi:10.1145/5925.5933.

42 Shaaban Ali Hoseinpour, Behrouz Afshar Nadjafi, and Seyed Taghi Akhavan Niaki. A bi-objective model for the firefighter problem to maximize fire protection with minimum firefighters. *Journal of Industrial Engineering and Management Studies*, 10(1):77–87, 2023.

43 Madhav Jha and Sofya Raskhodnikova. Testing and reconstruction of lipschitz functions with applications to data privacy. *SIAM Journal on Computing*, 42(2):700–731, 2013. doi:10.1137/110840741.

44 Nicolas Jozefowiez, Fred Glover, and Manuel Laguna. Multi-objective meta-heuristics for the traveling salesman problem with profits. *Journal of Mathematical Modelling and Algorithms*, 7(2):177–195, 2008. doi:10.1007/s10852-008-9080-2.

45 Özlem Karsu and Ayse Selin Kocaman. Towards the sustainable development goals: A bi-objective framework for electricity access. *Energy*, 216:119305, 2021. doi:10.1016/j.energy.2020.119305.

46 Carl Peter Keller. *Multiobjective routing through space and time: The MVP and TDVP problems*. PhD thesis, University of Western Ontario, 1985.

47 Carl Peter Keller and Michael Frank Goodchild. The multiobjective vending problem: a generalization of the travelling salesman problem. *Environment and Planning B: Planning and Design*, 15(4):447–460, 1988. doi:10.1068/b150447.

48 Subhash Khot. Ruling out ptas for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM Journal on Computing*, 36(4):1025–1071, 2006. doi:10.1137/S0097539705447037.

49 Samir Khuller, Anna Moss, and Joseph Seffi Naor. The budgeted maximum coverage problem. *Information Processing Letters*, 70(1):39–45, 1999. doi:10.1016/S0020-0190(99)00031-9.

50 Guy Kortsarz. On the hardness of approximating spanners. *Algorithmica*, 30:432–450, 2001. doi:10.1007/s00453-001-0021-y.

51 Guy Kortsarz and David Peleg. Generating sparse 2-spanners. *Journal of Algorithms*, 17(2):222–236, 1994. doi:10.1006/jagm.1994.1032.

52 Markus Leitner, Ivana Ljubić, and Markus Sinnl. A computational study of exact approaches for the bi-objective prize-collecting steiner tree problem. *INFORMS Journal on Computing*, 27(1):118–134, 2015. doi:10.1287/ijoc.2014.0614.

53 Libin Liang, Derek Cool, Nirmal Kakani, Guangzhi Wang, Hui Ding, and Aaron Fenster. Development of a multi-objective optimized planning method for microwave liver tumor ablation. In *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 110–118. Springer, 2019. doi:10.1007/978-3-030-32254-0_13.

54 Viswanath Nagarajan and R. Ravi. Approximation algorithms for distance constrained vehicle routing problems. *Networks*, 59(2):209–214, 2012. doi:10.1002/net.20435.

55 Christos H. Papadimitriou and Mihalis Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Proc. FOCS 2000*, pages 86–92, New Jersey, 2000. IEEE. doi:10.1109/SFCS.2000.892068.

56 Alice Paul, Daniel Freund, Aaron Ferber, David B. Shmoys, and David P. Williamson. Budgeted prize-collecting traveling salesman and minimum spanning tree problems. *Mathematics of Operations Research*, 45(2):576–590, 2020. doi:10.1287/moor.2019.1002.

57 Alice Paul, Daniel Freund, Aaron Ferber, David B. Shmoys, and David P. Williamson. Erratum to "budgeted prize-collecting traveling salesman and minimum spanning tree problems". *Mathematics of Operations Research*, 48(4):2304–2307, 2023. doi:10.1287/moor.2022.1340.

58 David Peleg. Proximity-preserving labeling schemes and their applications. In *WG 1999*, volume 1665 of *LNCS*, pages 30–41, Berlin, Heidelberg, 1999. Springer. doi:10.1007/3-540-46784-X_5.

59 David Peleg and Alejandro A. Schäffer. Graph spanners. *Journal of graph theory*, 13(1):99–116, 1989. doi:10.1002/jgt.3190130114.

60 David Peleg and Jeffrey D. Ullman. An optimal synchronizer for the hypercube. In *Proc. PODC 1987*, pages 77–85, New York, 1987. ACM. doi:10.1145/41840.41847.

61 Daniela Saban, Flavia Bonomo, and Nicolás E. Stier-Moses. Analysis and models of bilateral investment treaties using a social networks approach. *Physica A: Statistical Mechanics and its Applications*, 389(17):3661–3673, 2010. doi:10.1016/j.physa.2010.04.001.

62 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *Journal of the ACM*, 52(1):1–24, 2005. doi:10.1145/1044731.1044732.

63 George Tsaggouris and Christos Zaroliagis. Multiobjective optimization: Improved fptas for shortest paths and non-linear objectives with applications. *Theory of Computing Systems*, 45(1):162–186, 2009. doi:10.1007/s00224-007-9096-4.

64 Sergei Vassilvitskii and Mihalis Yannakakis. Efficiently computing succinct trade-off curves. *Theoretical Computer Science*, 348(2-3):334–356, 2005. doi:10.1016/j.tcs.2005.09.022.

65 Markus Wallinger, Daniel Archambault, David Auber, Martin Nöllenburg, and Jaakko Peltonen. Faster edge-path bundling through graph spanners. *Computer Graphics Forum*, 42(6):e14789, 2023. doi:10.1111/CGF.14789.

66 Kerui Weng. Distance constrained facility location problem. In *2009 IITA International Conference on Services Science, Management and Engineering*, pages 358–361, 2009. doi:10.1109/SSME.2009.66.

A Synopsis of Further Applications (cf. Table 1)

We briefly introduce the biobjective problems shown in Table 1. The objective functions are ordered in the same way they are mentioned here. Given a graph, the *Biobjective Quasi-Clique* problem [28, 29, 61], asks for a subgraph that simultaneously maximizes the number of contained nodes and the edge-density. Given an edge weighted graph, the *Biobjective Vehicle Routing* problem [20, 32] asks for a set of vehicle routes such that both the makespan (i.e., length of the longest route) and the number of required vehicles is minimized. Interestingly, combining [33, 54] even solves β-RelaxedRestrict(c) with $\beta = \min\{\log n, \frac{\log c}{\log\log c}\}$, where the accuracy improves with stricter budgets. Given a set of facilities F with opening costs, clients D, and a metric on $F \cup D$ for distances and travel costs, respectively, the *Biobjective Facility Location* problem [5], asks to open a subset of facilities such that both the radius (i.e., maximum distance of any client to its closest open facility) and the total costs, as induces by openings and traveling, are minimized. We also consider a variation without travel costs, where the opening costs are the first objective and the radius is the second. Given a universe of elements with profits and a family of sets over the universe with costs, the *Biobjective Price-Collecting Set Cover* problem asks for a selection of sets that both minimizes the total selection cost and maximizes the total profit gained by covering elements. It generalizes the (single-objective) *Budgeted Maximum Coverage* problem [49], where the profit is maximized given a cost limit, and finds application in, e.g., radiation treatment [53]. Given a graph with edge costs and node profits, the *Biobjective Price-Collecting Traveling Salesperson* problem [10, 44, 46, 47] asks for tours (that do not necessarily visit all nodes) that minimize the tour's cost and maximize the total profit gained from visiting nodes. Similarly, the *Biobjective Price-Collecting Steiner Tree* problem [45, 52] asks for a minimum-cost steiner tree that maximizes the total profit gained from visiting nodes. Note that, by [57], Theorems 4 and 5 of [56] only apply in the unrooted case. In [3], variations of the classic (single-objective) *Firefighter* problem are considered. Both maximizing the number of saved nodes and minimizing the number of required firefighters are deemed equally important objectives, motivating the *Biobjective Firefighter* problem [42].