FACHHOCHSCHULE TRIER
Hochschule für Technik, Wirtschaft und Gestaltung
University of Applied Sciences
Informatik

Informatik-Bericht Nr. 2010-3

Schriftenreihe Fachbereich Informatik, Fachhochschule Trier

Hardness and Approximability in Multi-Objective Optimization

Christian Glaßer* Christian Reitwießner† Heinz Schmitz‡ Maximilian Witek§

March 4, 2010

Abstract

We systematically study the hardness and the approximability of combinatorial multi-objective NP optimization problems (multi-objective problems, for short).

We define solution notions that precisely capture the typical algorithmic tasks in multi-objective optimization. These notions inherit polynomial-time Turing reducibility from multivalued functions, which allows us to compare the solution notions and to define corresponding NP-hardness notions. For both we prove reducibility and separation results.

Furthermore, we define approximative solution notions and investigate in which cases polynomial-time solvability translates from one to another notion. For problems where all objectives have to be minimized, approximability results translate from single-objective to multi-objective optimization such that the relative error degrades only by a constant factor. Such translations are not possible for problems where all objectives have to be maximized, unless P = NP.

As a consequence we see that in contrast to single-objective problems, where the solution notions coincide, the situation is more subtle for multiple objectives. So it is important to exactly specify the NP-hardness notion when discussing the complexity of multi-objective problems.

1 Introduction

Many technical, economical, natural- and socio-scientific processes contain multiple optimization objectives in a natural way. For instance, in logistics one is interested in routings that simultaneously minimize transportation costs and transportation time. For typical instances there does not exist a single solution that is optimal for both objectives, since they are conflicting. Instead one will encounter trade-offs between both objectives, i. e., some routings will be cheap, others will be fast. The *Pareto set* captures the notion of optimality in this setting. It consists of all solutions that are optimal in the sense that there is no solution that is strictly better. For decision makers the Pareto set is very useful as it reveals all trade-offs between all optimal solutions for the current instance.

In practice, multi-objective problems are often solved by turning the problem into a single-objective problem first and then solving that problem. This approach has the benefit that one can build on known techniques for single-objective problems. However, it comes along with the disadvantage that the translation of the problem significantly changes its nature such that sometimes the problem becomes harder to solve and sometimes certain optimal solutions are not found anymore. In general, single-objective problems cannot adequately represent multi-objective problems, so optimization of multi-objective problems is studied on its own. The

*Julius-Maximilians-Universität Würzburg, Germany. glasser@informatik.uni-wuerzburg.de
†Julius-Maximilians-Universität Würzburg, Germany. reitwiessner@informatik.uni-wuerzburg.de
‡Fachhochschule Trier, Germany. schmitz@informatik.fh-trier.de
§Julius-Maximilians-Universität Würzburg, Germany. witek@informatik.uni-wuerzburg.de

research area of multi-objective combinatorial optimization has its origins in the late 1980s and has become increasingly active since that time [EG02]. For a general introduction we refer to the survey by Ehrgott and Gandibleux [EG00] and the textbook by Ehrgott [Ehr05].

Typically, the exact solution of a multi-objective problem is not easier than the exact solution of an underlying single-objective problem. Therefore, polynomial-time approximation with performance guarantee is a reasonable approach also to multi-objective problems. In this regard, Papadimitriou and Yannakakis [PY00] show an important result: Every Pareto set has a $(1+\varepsilon)$-approximation of size polynomial in the size of the instance and $1/\varepsilon$. Hence, even though a Pareto set might be an exponentially large object, there always exists a polynomial-sized approximative set. This clears the way for a general investigation of the approximability of Pareto sets. However, complexity issues raised by multi-objective problems have not been addressed systematically yet [PY00]. We consider our paper as a first step to a systematic investigation of hardness and approximability of multi-objective problems. Our contribution is as follows:

Solution Notions (Section 3): We define several notions that capture reasonable algorithmic tasks for computing *optimal* solutions of multi-objective problems. On the technical side, we see that these notions can be uniformly and precisely described by refinements of total, multivalued functions. This yields the suitable concept of polynomial-time Turing reducibility for solution notions. It turns out that the relationships shown in Figure 1 hold for arbitrary multi-objective problems.

NP-*Hardness Notions* (Section 4): Solution notions for multi-objective problems induce corresponding NP-hardness notions for these problems. In Table 1 we provide examples of multi-objective problems that are hard with respect to some notion, but polynomial-time solvable with respect to some other notion. So in contrast to the single-objective case where all notions coincide, we see a more subtle picture in case of multiple objectives: our separation results show that NP-hardness notions in fact differ, unless P = NP. As a consequence, we suggest to exactly specify the solution notion when discussing the complexity of multi-objective problems and when comparing problems in these terms.

Approximation Notions (Section 5): We also define and investigate various approximative solution notions for multi-objective problems. As a summary, Figure 2 shows for arbitrary multi-objective problems in which cases polynomial-time solvability of one such notion implies polynomial-time solvability of another notion, and what quality of approximation can be preserved at least. Moreover, we reveal a significant dichotomy between approximation of minimization and maximization problems in this context. For problems where all objectives have to be minimized, approximability results translate from single-objective to multi-objective optimization such that the relative error degrades only by a constant factor (the number of objectives). With this general result we provide a procedure how to translate (already known) single-objective approximation results to the multi-objective case. Applications to some example problems are stated in Table 2. In contrast to this result we prove that such translations are not possible for problems where all objectives have to be maximized, unless P = NP.

2 Preliminaries

Let $k \geq 1$. A *combinatorial k-objective* NP *optimization problem* (*k-objective problem*, for short) is a tuple $(S, f, \leftarrow)$ where

- $S\colon \mathbb{N} \rightarrow 2^{\mathbb{N}}$ maps an instance $x \in \mathbb{N}$ to the set of feasible solutions for this instance, denoted as $S^x \subseteq \mathbb{N}$. There must be some polynomial p such that for every $x \in \mathbb{N}$ and every $s \in S^x$ it holds that $|s| \leq p(|x|)$ and the set $\{(x, s) \mid x \in \mathbb{N}, s \in S^x\}$ must be polynomial-time decidable.

- $f\colon \{(x,s) \mid x \in \mathbb{N}, s \in S^x\} \to \mathbb{N}^k$ maps an instance $x \in \mathbb{N}$ and a solution $s \in S^x$ to its value, denoted by $f^x(s) \in \mathbb{N}^k$. f must be polynomial-time computable.
- $\leftarrow \subseteq \mathbb{N}^k \times \mathbb{N}^k$ is the partial order relation specifying the direction of optimization. It must hold that $(a_1,\dots,a_k) \leftarrow (b_1,\dots,b_k) \iff a_1 \leftarrow_1 b_1 \wedge \dots \wedge a_k \leftarrow_k b_k$, where $\leftarrow_i$ is $\leq$ if the i-th objective is minimized, and $\leftarrow_i$ is $\geq$ if the i-th objective is maximized.

For instances and solutions we relax the restriction to integers and allow other objects (e. g., graphs) where a suitable encoding is assumed, possibly setting $S^x = \emptyset$ if x is not a valid code. We write $\leq$ and $\geq$ also for their multidimensional variants, i. e., $\leq$ is used as the partial order $\leftarrow$ where $\leftarrow_i = \leq$ for all i.

Our notation allows concise definitions of multi-objective problems. We exemplify this by defining two well-known problems on labeled graphs. An $\mathbb{N}^k$-*node-labeled* (resp., $\mathbb{N}^k$-*edge-labeled*) graph is a triple $G = (V,E,l)$ such that (V,E) is a graph and $l\colon V \to \mathbb{N}^k$ (resp., $l\colon E \to \mathbb{N}^k$) is a total function.

Definition 2.1 (k-Objective Minimum Matching)**.**

> ***k*-MM** $= (S,f,\leq)$ *where instances are* $\mathbb{N}^k$*-edge-labeled graphs* $G = (V,E,l)$, $S^G = \{M \mid M \subseteq E$ *is a perfect matching on* $G\}$, *and* $f^G(M) = \sum_{e\in M} l(e)$.

Definition 2.2 (2-Objective Minimum Traveling Salesman)**.**

> **2-TSP** $= (S,f,\leq)$ *where instances are* $\mathbb{N}^2$*-edge-labeled graphs* $G = (V,E,l)$, $S^G = \{H \mid H \subseteq E$ *is a Hamiltonian circuit in* $G\}$, *and* $f^G(H) = \sum_{e\in H} l(e)$.

The superscript x of f and S can be omitted if it is clear from context. The projection of f^x to the i-th component is denoted as f^x_i where $f^x_i(s) = v_i$ if $f^x(s) = (v_1,\dots,v_k)$. Furthermore, the order relation $\leftarrow$ obtained from $\leftarrow_1,\dots,\leftarrow_k$ is also written as $(\leftarrow_1,\dots,\leftarrow_k)$. If $a \leftarrow b$ we say that *a weakly dominates b* (i. e., a is at least as good as b). If $a \leftarrow b$ and $a \neq b$ we say that *a dominates b*. Note that $\leftarrow$ always points in the direction of the better value. If f and x are clear from the context, then we extend $\leftarrow$ to combinations of values and solutions. So we can talk about weak dominance between solutions, and we write $s \leftarrow t$ if $f^x(s) \leftarrow f^x(t)$, $s \leftarrow c$ if $f^x(s) \leftarrow c$, and so on, where $s,t \in S^x$ and $c \in \mathbb{N}^k$. Furthermore, we define $\mathrm{opt}_\leftarrow\colon 2^{\mathbb{N}^k} \to 2^{\mathbb{N}^k}$, $\mathrm{opt}_\leftarrow(M) = \{y \in M \mid \forall z \in M[z \leftarrow y \Rightarrow z = y]\}$ as a function that maps sets of values to sets of optimal values. The operator $\mathrm{opt}_\leftarrow$ is also applied to sets of solutions $S' \subseteq S^x$ as $\mathrm{opt}_\leftarrow(S') = \{s \in S' \mid f^x(s) \in \mathrm{opt}_\leftarrow(f^x(S'))\}$. If even $\leftarrow$ is clear from the context, we write $S^x_{\mathrm{opt}} = \mathrm{opt}_\leftarrow(S^x)$ and $\mathrm{opt}_i(S') = \{s \in S' \mid f^x_i(s) \in \mathrm{opt}_{\leftarrow_i}(f^x_i(S'))\}$.

For approximations we need to relax the notion of dominance by a factor of α. For any real $a \geq 1$ define $u \overset{a}{\leq} v \iff u \leq a \cdot v$ and $u \overset{a}{\geq} v \iff a \cdot u \geq v$. Fix some $\leftarrow = (\leftarrow_1,\dots,\leftarrow_k)$ where $\leftarrow_i \in \{\leq,\geq\}$, let $p = (p_1,\dots,p_k), q = (q_1,\dots,q_k) \in \mathbb{N}^k$, and let $\alpha = (a_1,\dots,a_k) \in \mathbb{R}^k$ where $a_1,\dots,a_k \geq 1$. We say that p *weakly* α*-dominates* q, $p \overset{\alpha}{\leftarrow} q$ for short, if $p_i \overset{a_i}{\leftarrow}_i q_i$ for $1 \leq i \leq k$. For all $p,q,r \in \mathbb{N}^k$ it holds that $p \overset{\alpha}{\leftarrow} p$, and $p \overset{\alpha}{\leftarrow} q \overset{\beta}{\leftarrow} r \implies p \overset{\alpha\cdot\beta}{\leftarrow} r$, where $\alpha\cdot\beta$ is the component-wise multiplication. Again we extend $\overset{\alpha}{\leftarrow}$ to combinations of values and solutions, if f and x are clear from the context.

Let A and B be sets. $\mathcal{F}$ is a *multivalued function* from A to B, if $\mathcal{F} \subseteq A \times B$. The *set of values* of x is set-$\mathcal{F}(x) = \{y \mid (x,y) \in \mathcal{F}\}$. $\mathcal{F}$ is called *total*, if for all x, set-$\mathcal{F}(x) \neq \emptyset$.

In order to compare solution notions of optimization problems we need an appropriate reducibility notion. All solution notions $\mathcal{F}$ considered in this paper have in common that each instance x specifies a non-empty set of suitable outputs

$$\text{set-}\mathcal{F}(x) = \{y \mid y \text{ solves } x \text{ in terms of solution notion } \mathcal{F}\}.$$

In this sense, a solution notion $\mathcal{F}$ is a total multivalued function that maps an instance x to all $y \in \text{set-}\mathcal{F}(x)$. Therefore, solution notions can be compared by means of a reducibility for total multivalued functions. We use Selman's [Sel94] definition of polynomial-time Turing reducibility for multivalued functions, restricted to *total* multivalued functions. First, let us specify how a Turing transducer [BLS84] uses a total function f as oracle: If the transducer writes a query q to the query tape and changes into the query state, then at the next step, the machine is in the answer state, the query tape is empty, and the content of the answer tape is $f(q)$, where the head is on the left-most symbol of $f(q)$. A total function f is a *refinement* of a total multivalued function $\mathcal{F}$, if for all x, $f(x) \in \text{set-}\mathcal{F}(x)$. A total multivalued function $\mathcal{F}$ is *polynomial-time Turing reducible* to a total multivalued function $\mathcal{G}$, $\mathcal{F} \leq_{\mathrm{T}}^{\mathrm{p}} \mathcal{G}$, if there exists a deterministic, polynomial-time-bounded oracle Turing transducer M such that for every refinement g of $\mathcal{G}$ it holds that M with g as oracle computes a total function that is a refinement of $\mathcal{F}$. Note that the oracle model ensures that $\leq_{\mathrm{T}}^{\mathrm{p}}$ is transitive, even if the lengths of the elements in set-$\mathcal{F}(x)$ are not polynomially bounded in $|x|$.

The decision problem of a set A can be considered as computing the characteristic function χ_A, which in turn is a total (multivalued) function. In this way, the polynomial-time Turing reducibility defined above can also be applied to decision problems.

A solution notion $\mathcal{F}$ is called *polynomial-time solvable*, if there is a total, polynomial-time computable function f such that f is a refinement of $\mathcal{F}$. A solution notion $\mathcal{F}$ is called NP-*hard*, if all problems in NP are polynomial-time Turing-reducible to $\mathcal{F}$.

3 Multi-Objective Solution Notions

For a k-objective problem $\mathcal{O} = (S, f, \leftarrow)$ we discuss several reasonable concepts of "solving $\mathcal{O}$". We investigate their relationships and conclude this section with a taxonomy of these concepts.

Apparently a dominated solution s is not optimal for $\mathcal{O}$, since solutions exist that are at least as good as s in all objectives and better than s in at least one objective. So we are only interested in non-dominated solutions, which are called *(Pareto-)optimal solutions.* Note that the set S_{opt}^x of non-dominated solutions may contain several solutions with identical values. Since these solutions cannot be distinguished, it suffices to find one solution for each optimal value, as it is usual in single-objective optimization. This motivates the following definition.

E-$\mathcal{O}$ **Every-optimum notion**
Compute a set of optimal solutions that generate all optimal values.
Input: instance x
Output: some $S' \subseteq S_{\text{opt}}^x$ such that $f^x(S') = f^x(S_{\text{opt}}^x)$

Although E-$\mathcal{O}$ formalizes the canonical notion of solving multi-objective problems, this is far too ambitious in many cases, since every set S' can be of exponential size. We call $\mathcal{O}$ *polynomially bounded* if there is some polynomial p such that $\#f^x(S_{\text{opt}}^x) \leq p(|x|)$ for all x. If $\mathcal{O}$ is not polynomially bounded, then E-$\mathcal{O}$ is not polynomial-time solvable. The earlier defined problems 2-MM and 2-TSP are examples that show this effect. So E-$\mathcal{O}$ is infeasible in general, and hence more restricted concepts of solving multi-objective problems are needed. This brings us to the following specifications for $\mathcal{O}$.

A-$\mathcal{O}$ **Arbitrary-optimum notion**

Compute an arbitrary optimal solution.
Input: instance x
Output: some $s \in S^x_{\text{opt}}$ or report that $S^x = \emptyset$

S-$\mathcal{O}$ **Specific-optimum notion**

Compute an optimal solution that weakly dominates a given cost vector.
Input: instance x, cost vector $c \in \mathbb{N}^k$
Output: some $s \in S^x_{\text{opt}}$ with $f^x(s) \leftarrow c$ or report that there is no such s

D-$\mathcal{O}$ **Dominating-solution notion**

Compute a solution that weakly dominates a given cost vector.
Input: instance x, cost vector $c \in \mathbb{N}^k$
Output: some $s \in S^x$ with $f^x(s) \leftarrow c$ or report that there is no such s

If no additional information is available (including no further criteria, no prior knowledge, and no experience by decision makers), then it is not plausible to distinguish non-dominated solutions. In these cases it suffices to consider A-$\mathcal{O}$, since all elements in S^x_{opt} are "equally optimal". The notion S-$\mathcal{O}$ additionally allows us to specify the minimal quality c that an optimal solution s must have. With D-$\mathcal{O}$ we relax the constraint that s must be optimal.

There exist several well-established approaches that turn $\mathcal{O}$ into a single-objective problem first and then treat it with methods known from single-objective optimization. The following definitions are motivated by such methods. Later we will show that these approaches differ with respect to their computational complexity (cf. Figure 1 and Table 1). Note that we consider W-$\mathcal{O}$ only for multi-objective problems where either all objectives have to be minimized or all have to be maximized (for other problems it is not clear whether the weighted sum should be minimized or maximized).

W-$\mathcal{O}$ **Weighted-sum notion** (only if all objectives are minimized or all are maximized)

Single-objective problem that weights the objectives in a given way.
Input: instance x, weight vector $\omega \in \mathbb{N}^k$
Output: some $s \in S^x$ that optimizes $\sum_{i=1}^k \omega_i f^x_i(s)$ or report that $S^x = \emptyset$

C$_\text{i}$-$\mathcal{O}$ **Constraint notion for the *i*-th objective**

Single-objective problem that optimizes the i-th objective while respecting constraints on the remaining objectives.
Input: instance x, constraints $b_1, \dots, b_{i-1}, b_{i+1}, \dots, b_k \in \mathbb{N}$
Output: for $S^x_{\text{con}} = \{s \in S^x \mid f^x_j(s) \leftarrow_j b_j \text{ for all } j \neq i\}$ return some $s \in \text{opt}_i(S^x_{\text{con}})$ or report that $S^x_{\text{con}} = \emptyset$

L-$\mathcal{O}$ **Lexicographical notion for a fixed order of objectives**

Single-objective problem with a fixed order of objectives (here $1, 2, \dots, k$).
Input: instance x
Output: some $s \in \text{opt}_k(\dots(\text{opt}_2(\text{opt}_1(S^x)))\dots)$ or report that $S^x = \emptyset$

Strictly speaking, W-$\mathcal{O}$, C$_\text{i}$-$\mathcal{O}$, and L-$\mathcal{O}$ are not only solution notions for $\mathcal{O}$, but in fact they are single-objective problems. In the literature, L-$\mathcal{O}$ is also known as *hierarchical optimization*.

Moreover, W-$\mathcal{O}$ is a particular *normalization approach*, since a norm is used to aggregate several cost functions into one. We will get back to general normalized approaches in section 5.3.

Proposition 3.1. *Let $\mathcal{O}$ be a k-objective problem. If* E-$\mathcal{O}$ *is polynomial-time solvable, then* A-$\mathcal{O}$*,* S-$\mathcal{O}$*,* D-$\mathcal{O}$*,* W-$\mathcal{O}$*,* L-$\mathcal{O}$*,* $\mathrm{C_1}$-$\mathcal{O}, \dots, \mathrm{C_k}$-$\mathcal{O}$ *are polynomial-time solvable.*

Moreover, if $\mathcal{O}$ is a single-objective problem, then all notions defined so far are polynomial-time Turing equivalent.

E-$\mathcal{O}$ plays a special role among the solution notions for $\mathcal{O}$, since solutions of E-$\mathcal{O}$ are typically of exponential size. So in general, E-$\mathcal{O}$ is not polynomial-time Turing reducible to any of the other notions. On the other hand, polynomial-time Turing reductions to E-$\mathcal{O}$ are problematic as well, since answers to oracle queries can be exponentially large which makes the reduction very sensitive to encoding issues (as the reduction machine can only read the left-most polynomial-size part of the answer). Therefore, we will not compare E-$\mathcal{O}$ with the other notions by means of polynomial-time Turing reductions, and we will not consider the NP-hardness of E-$\mathcal{O}$. However, in some sense E-$\mathcal{O}$ is covered by D-$\mathcal{O}$ (resp., S-$\mathcal{O}$), since the latter can be considered as some special polynomial-time oracle access to E-$\mathcal{O}$, i. e., D-$\mathcal{O}$ (resp., S-$\mathcal{O}$) can be used in a binary search manner to find solutions for arbitrary optimal values. In Section 5, E-$\mathcal{O}$ will become important again in the context of approximate solutions.

Next we show that the remaining notions are closely related (see Figure 1 for a summary). SAT denotes the NP-complete set of all satisfiable Boolean formulas.

Theorem 3.2. *Let $\mathcal{O} = (S, f, \leftarrow)$ be some k-objective problem.*

1. A-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ L-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ S-$\mathcal{O}$
2. S-$\mathcal{O} \equiv^{\mathrm{p}}_{\mathrm{T}}$ D-$\mathcal{O} \equiv^{\mathrm{p}}_{\mathrm{T}}$ $\mathrm{C_1}$-$\mathcal{O} \equiv^{\mathrm{p}}_{\mathrm{T}}$ $\mathrm{C_2}$-$\mathcal{O} \equiv^{\mathrm{p}}_{\mathrm{T}} \dots \equiv^{\mathrm{p}}_{\mathrm{T}}$ $\mathrm{C_k}$-$\mathcal{O}$
3. D-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ SAT
4. L-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ W-$\mathcal{O}$ *and* W-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ SAT *if all objectives have to be minimized (resp., maximized)*

Proof.

1. Any solution of L-$\mathcal{O}$ is an optimal solution and thus solves A-$\mathcal{O}$. To solve L-$\mathcal{O}$ we use S-$\mathcal{O}$ to perform a binary search that respects the priority of objectives given in L-$\mathcal{O}$ (i. e., we first optimize the objectives with the higher priority and proceed by optimizing the lower prioritized objectives, while forcing the optimal values for objectives with higher priority).

2. First observe that D-$\mathcal{O} \equiv^{\mathrm{p}}_{\mathrm{T}}$ S-$\mathcal{O}$, since a solution to S-$\mathcal{O}$ is also a solution to D-$\mathcal{O}$, whereas a binary search on D-$\mathcal{O}$ also solves S-$\mathcal{O}$. Now, suppose we want to solve $\mathrm{C_i}$-$\mathcal{O}$. A binary search over objective i that keeps the other objectives fixed to their constrained values shows $\mathrm{C_i}$-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ D-$\mathcal{O}$. On the other hand, optimizing objective i with constraints $b_j = c_j$ for all $j \neq i$ where $c = (c_1, \dots, c_k)$ is the input cost vector for D-$\mathcal{O}$ shows D-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ $\mathrm{C_i}$-$\mathcal{O}$.

3. By the definition of k-objective problems, the set

$$B_{\mathcal{O}} = \{(x, s', c) \in \mathbb{N} \times \mathbb{N} \times \mathbb{N}^k \mid \exists s \in S^x[s \leftarrow c \text{ and } s \text{ has prefix } s']\}$$

is contained in NP. A single query to $B_{\mathcal{O}}$ with the empty prefix checks whether there is a solution $s \in S^x$ with $s \leftarrow c$, which, in turn, can then be obtained by a binary search that sequentially sets every bit of s by querying $B_{\mathcal{O}}$. This shows D-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}} B_{\mathcal{O}}$ and together with $B_{\mathcal{O}} \leq^{\mathrm{p}}_{\mathrm{T}}$ SAT we get D-$\mathcal{O} \leq^{\mathrm{p}}_{\mathrm{T}}$ SAT.

4. Since f is polynomial-time computable, there is some polynomial p such that $|f_i^x(s)| \leq p(|x|)$, hence $f_i^x(s) \leq 2^{p(|x|)}$ for all x, i and $s \in S^x$. Given some instance x and a fixed order of objectives we use the weight $(2^{p(|x|)})^{k-1}$ for the objective with the highest priority, $(2^{p(|x|)})^{k-2}$ for the objective with the second highest priority and so on. This shows L-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}}$ W-$\mathcal{O}$.

   To show W-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}}$ SAT, observe that W-$\mathcal{O}$ is equivalent to the single-objective problem $\mathcal{O}' = (S, f', \leftarrow_1)$, where f' is the weighted sum over all objectives of $\mathcal{O}$, hence W-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}}$ A-$\mathcal{O}'$. But we already know A-$\mathcal{O}' \leq_{\mathrm{T}}^{\mathrm{p}}$ D-$\mathcal{O}'$ and D-$\mathcal{O}' \leq_{\mathrm{T}}^{\mathrm{p}}$ SAT, hence we get W-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}}$ SAT. □

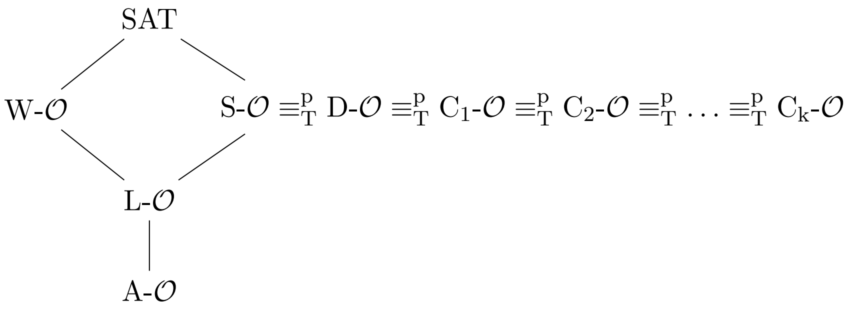
SAT
W-$\mathcal{O}$
S-$\mathcal{O} \equiv_{\mathrm{T}}^{\mathrm{p}}$ D-$\mathcal{O} \equiv_{\mathrm{T}}^{\mathrm{p}}$ C$_1$-$\mathcal{O} \equiv_{\mathrm{T}}^{\mathrm{p}}$ C$_2$-$\mathcal{O} \equiv_{\mathrm{T}}^{\mathrm{p}} \ldots \equiv_{\mathrm{T}}^{\mathrm{p}}$ C$_\mathrm{k}$-$\mathcal{O}$
L-$\mathcal{O}$
A-$\mathcal{O}$

Figure 1: Polynomial-time turing reducibility among different solution notions for any multi-objective problem $\mathcal{O}$. Corresponding separations are shown in Section 4. Note that W-$\mathcal{O}$ is only defined if all objectives are minimized or all are maximized.

We will see that Theorem 3.2 is complete in the sense that no further reductions among the solution notions are possible in general and the complexity of different notions can be separated, unless P = NP (cf. Corollary 4.11). Therefore, even if multi-objective problems are not polynomially bounded it is worthwhile to classify them according to the taxonomy given in Figure 1.

Further research: Each solution notion for a multi-objective problem $\mathcal{O}$ gives rise to a total multivalued function (see Section 2). To obtain more detailed insights in the complexity of different solution notions one may draw connections to classical complexity theory of partial multivalued functions, starting with the seminal work of Selman [Sel94]. We expect to see a rich structure of multivalued function classes that are defined on basis of different solution notions. At this point, more research is needed.

4 Multi-Objective NP-Hardness

Each solution notion for a multi-objective problem induces a corresponding NP-hardness notion. In this section we study the relationships between these NP-hardness notions and present separation results which imply the strictness of the reductions shown in Theorem 3.2 and Figure 1.

From Theorem 3.2 we obtain the following relationships between NP-hardness notions.

Theorem 4.1. *Let $\mathcal{O} = (S, f, \leftarrow)$ be some k-objective problem.*

1. A-$\mathcal{O}$ NP-*hard* $\Longrightarrow$ L-$\mathcal{O}$ NP-*hard* $\Longrightarrow$ S-$\mathcal{O}$ NP-*hard and* W-$\mathcal{O}$ NP-*hard*
2. S-$\mathcal{O}$ NP-*hard* $\Longleftrightarrow$ D-$\mathcal{O}$ NP-*hard* $\Longleftrightarrow$ C$_\mathrm{i}$-$\mathcal{O}$ NP-*hard*

We show that no further implications hold between the NP-hardness of the solution notions, unless P = NP. Such differences concerning the NP-hardness are not unusual. Below we give several examples of natural problems that are NP-hard with respect to one notion and that are polynomial-time solvable with respect to another one. This shows the importance of an exact specification of the NP-hardness notion that is used when discussing the complexity of multi-objective problems.

The following problem is one of the most studied 2-objective scheduling problems in the literature [LV93].

Definition 4.2 (Minimimum Lateness and Weighted Flowtime Scheduling)**.**

2-LWF $= (S, f, \leq)$ *where instances are triples* (P, D, W) *such that:*

- $P = (p_1, \ldots, p_n) \in \mathbb{N}^n$ *are processing times*
- $D = (d_1, \ldots, d_n) \in \mathbb{N}^n$ *are due dates*
- $W = (w_1, \ldots, w_n) \in \mathbb{N}^n$ *are weights*
- $S^{(P,D,W)} = \{\pi \mid \pi$ *is a permutation representing the schedule* $p_{\pi(1)}, \ldots, p_{\pi(n)}\}$
- $f^{(P,D,W)}(\pi) = (L_{\max}, \sum_{j=1}^{n} w_j C_j)$ *where*
  - *The* completion time *of job* j *is* $C_j = \sum_{i:\pi(i)\leq\pi(j)} p_i$.
  - *The* maximum lateness *is* $L_{\max} = \max\{C_j - d_j \mid 1 \leq j \leq n\}$.
  - *The* weighted flowtime *is* $\sum_{j=1}^{n} w_j C_j$.

2-LF = 2-LWF *where all weights are* 1*, i. e.,* $W = (1, \ldots, 1)$.

Besides scheduling problems we use multi-objective problems that aim at diophantine equations, shortest paths, and minimum spanning trees.

Definition 4.3 (Minimum Quadratic Diophantine Equations)**.**

2-QDE $= (S, f, \leq)$ *where instances are triples* $(a, b, c) \in \mathbb{N}^3$,
$S^{(a,b,c)} = \{(x, y) \in \mathbb{N}^2 \mid ax^2 + by^2 - c \geq 0\}$*, and* $f^{(a,b,c)}(x, y) = (x^2, y^2)$.

Definition 4.4 (2-Objective Shortest Path)**.**

2-SP $= (S, f, \leq)$ *where instances are* $\mathbb{N}^2$*-edge-labeled graphs* $G = (V, E, l)$ *and two distinct vertices* $s, t \in V$*,* $S^{(G,s,t)} = \{P \mid P \subseteq E$ *is a path that connects* s *and* t *in* $G\}$*, and* $f^{(G,s,t)}(P) = \sum_{e\in P} l(e)$.

Definition 4.5 (2-Objective Minimum Spanning Tree)**.**

2-MST $= (S, f, \leq)$ *where instances are* $\mathbb{N}^2$*-edge-labeled graphs* $G = (V, E, l)$*,*
$S^G = \{T \mid T \subseteq E$ *is a spanning tree of* $G\}$*, and* $f^G(T) = \sum_{e\in T} l(e)$.

The following propositions clarify the complexity of all solution notions for 2-MM, 2-SP, 2-MST, 2-QDE, 2-LWF, and 2-TSP. The results are summarized in Table 1.

Proposition 4.6. *Let* $\mathcal{O} \in \{$2-MM, 2-SP, 2-MST$\}$.

1. W-$\mathcal{O}$ *is polynomial-time solvable.*

2. C_1-$\mathcal{O}$ *is* NP-*hard.*

Proof. Consider $\mathcal{O}$ = 2-MM.

1. W-$\mathcal{O}$ is the single-objective problem that on input of an $\mathbb{N}^2$-edge-labeled graph $G = (V, E, l)$ and some weight vector (ω_1, ω_2) searches some perfect matching M of G such that $\sum_{e \in M}(\omega_1 \cdot l_1(e) + \omega_2 \cdot l_2(e))$ is minimal. In order to reduce W-$\mathcal{O}$ to Minimum Matching, we transform G to the $\mathbb{N}^1$-edge-labeled graph $G' = (V, E, l')$ such that $l'(e) = \omega_1 \cdot l_1(e) + \omega_2 \cdot l_2(e)$ for all $e \in E$. Minimum Matching is polynomial-time solvable, and its solution for G' yields an optimal solution of W-$\mathcal{O}$ for G and (ω_1, ω_2).

2. It is NP-hard to find the optimal value for one objective with a constraint on the other [PY82], hence the C_i-problems of 2-MM are NP-hard.

For 2-SP and 2-MST the proposition can be shown analogously, where the NP-hardness follows from [Han79, PY82]. □

Proposition 4.7. *Let* $\mathcal{O}$ = 2-QDE.

1. D-$\mathcal{O}$ *is polynomial-time solvable.*

2. W-$\mathcal{O}$ *is* NP-*hard.*

Proof. Consider $\mathcal{O}$ = 2-QDE.

1. C_1-$\mathcal{O}$ is the single-objective problem that on input of $a, b, c, b_2 \in \mathbb{N}$ searches for the smallest $x \in \mathbb{N}$ such that $\exists y \in \mathbb{N}[y^2 \leq b_2 \wedge ax^2 + by^2 - c \geq 0]$. Note that in the case of b_2 not being a square, replacing it by the greatest square smaller than b_2 does not change the problem. So we can assume b_2 to be a square. If $c \leq bb_2$, we obviously have $x = 0$, and if $c > bb_2$ and $a = 0$, there is no solution. Otherwise, we have $c > bb_2$ and $a > 0$, which directly implies $x = \left\lceil \sqrt{\frac{c - bb_2}{a}} \right\rceil$. Since all computations can be carried out and all cases can be distinguished efficiently, C_1-$\mathcal{O}$ and D-$\mathcal{O}$ are polynomial-time solvable.

2. The set QDE $= \{(a, b, c) \in \mathbb{N} \mid \exists x, y \in \mathbb{N}[ax^2 + by^2 - c = 0]\}$ is NP-complete [MA78]. We reduce QDE to W-$\mathcal{O}$. For given (a, b, c) we solve W-$\mathcal{O}$ for the weight vector $w = (a, b)$. If no solution is found, then $S^{(a,b,c)} = \emptyset$ and hence $(a, b, c) \notin$ QDE. Otherwise, let (x, y) be the solution of W-$\mathcal{O}$, i. e., $x, y \in \mathbb{N}$ such that $ax^2 + by^2 - c \geq 0$ and $ax^2 + by^2$ is minimal. It follows that $(a, b, c) \in$ QDE if and only if $ax^2 + by^2 - c = 0$. This shows the NP-hardness of W-$\mathcal{O}$. □

Proposition 4.8 ([HvdV90])**.** *For* $\mathcal{O}$ = 2-LF*,* E-$\mathcal{O}$ *is polynomial-time solvable.*

Proof. Hoogeveen and van de Velde [HvdV90] show that 2-LF is polynomially bounded and that there exists a polynomial-time algorithm that on input x computes some $S \subseteq S^x_{\mathrm{opt}}$ such that $f^x(S) = f^x(S^x_{\mathrm{opt}})$ [HvdV90, Theorem 8], [Hoo92, page 15]. □

Proposition 4.9 ([Bak74, Hoo92])**.** *Let* $\mathcal{O}$ = 2-LWF*, let* L_1-$\mathcal{O}$ *be the solution notion that first minimizes the maximum lateness, and* L_2-$\mathcal{O}$ *the notion that first minimizes the weighted flowtime.*

1. L_1-$\mathcal{O}$ *is* NP-*hard.*

2. L_2-$\mathcal{O}$ *is polynomial-time solvable.*

Proof. 1. Scheduling problems are often denoted by the three-field notation scheme $\alpha|\beta|\gamma$ introduced by Graham et al. [GLLK79], where α describes the machine environment, β the job constraints, and γ the objective function. In this notation, the problem L_1-$\mathcal{O}$ is written as $1||F_h(L_{\max}, \sum w_j C_j)$, where the F_h indicates that the optimization is hierarchical such that $L_{\max}$ is the primary and $\sum w_j C_j$ the secondary objective. Hoogeveen [Hoo92, Theorem 11] shows the NP-hardness of $1||F_h(L_{\max}, \sum w_j C_j)$.

2. The problem $1||F_h(\sum w_j C_j, L_{\max})$ is solved in time $O(n \log n)$ by sequencing the jobs in nondecreasing order of the ratios p_i/w_i (which minimizes the weighted flowtime) such that ties are broken by sequencing the jobs in nondecreasing order of their due dates (which minimizes the maximum lateness) [Bak74], [Hoo92, page 17]. Hence L_2-$\mathcal{O}$ is polynomial-time solvable. □

Proposition 4.10. *For* $\mathcal{O} = 2$-TSP *it holds that* A-$\mathcal{O}$ *is* NP-*hard.*

Proof. Let $G_1 = (V, E, l_1)$ be an instance of single-objective TSP (1-TSP), i. e., an $\mathbb{N}$-edge-labeled graph. $G_2 = (V, E, l_2)$ is an instance of 2-TSP where $l_2(e) = (l_1(e), l_1(e))$ for $e \in E$. A tour is optimal for G_2 if and only if it is optimal for G_1. So A-$\mathcal{O}$ is NP-hard. □

Problem $\mathcal{O}$	A-$\mathcal{O}$	L_1-$\mathcal{O}$	L_2-$\mathcal{O}$	W-$\mathcal{O}$	S-$\mathcal{O}$,D-$\mathcal{O}$,$\mathrm{C_i}$-$\mathcal{O}$	Ref.
2-LF	P	P	P	P	P	Prop. 4.8
2-MM	P	P	P	P	NP-hard	Prop. 4.6
2-SP	P	P	P	P	NP-hard	Prop. 4.6
2-MST	P	P	P	P	NP-hard	Prop. 4.6
2-QDE	P	P	P	NP-hard	P	Prop. 4.7
2-LWF	P	NP-hard	P	NP-hard	NP-hard	Prop. 4.9
2-TSP	NP-hard	NP-hard	NP-hard	NP-hard	NP-hard	Prop. 4.10

Table 1: Separation of NP-hardness notions for multi-objective problems. ‘P’ indicates that this solution notion for the problem is polynomial-time solvable. $\mathrm{L_i}$-$\mathcal{O}$ denotes the lexicographical problem where the i-th objective is the primary one.

As a consequence, under the assumption P ≠ NP we can prove the strictness of the Turing-reduction order shown in Theorem 3.2 and Figure 1.

Corollary 4.11. *If* P ≠ NP *then there exist* 2*-objective problems* $\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3, \mathcal{O}_4, \mathcal{O}_5$ *where all objectives have to be minimized such that the following holds.*

1. L-$\mathcal{O}_1 \not\leq^{\mathrm{p}}_{\mathrm{T}}$ A-$\mathcal{O}_1$

2. W-$\mathcal{O}_2 \not\leq^{\mathrm{p}}_{\mathrm{T}}$ L-$\mathcal{O}_2$

3. D-$\mathcal{O}_3 \not\leq^{\mathrm{p}}_{\mathrm{T}}$ L-$\mathcal{O}_3$

4. D-$\mathcal{O}_4 \not\leq^{\mathrm{p}}_{\mathrm{T}}$ W-$\mathcal{O}_4$

5. W-$\mathcal{O}_5 \not\leq^{\mathrm{p}}_{\mathrm{T}}$ D-$\mathcal{O}_5$

Further research: The solution notions A-$\mathcal{O}$, L-$\mathcal{O}$, W-$\mathcal{O}$, S-$\mathcal{O}$, D-$\mathcal{O}$, $\mathrm{C_i}$-$\mathcal{O}$, and E-$\mathcal{O}$ aim at the computation of an optimal *solution.* If one is interested in the hardness of a multi-objective problem, then it makes sense to consider also the computation of *values* of optimal solutions. For instance, Ehrgott [Ehr05] calls a multi-objective problem NP-complete, if the related decision problem is NP-complete. The latter problem is the question of whether there exists a solution that satisfies given contraints, which is polynomial-time Turing equivalent to the computation of the values of optimal solutions. Clearly, hardness with respect to value computations implies hardness with respect to solution computations, but the converse seems not to be true. At this point, further research is neccessary.

5 Multi-Objective Approximation

We define several approximation notions for multi-objective problems and study their relationships. Moreover, we show that if all objectives have to be minimized, then approximability results translate from single-objective to multi-objective optimization (cf. Table 2 for examples). In contrast, such translations are not possible for problems where all objectives have to be maximized, unless P = NP. Figure 2 summarizes the results obtained in this section.

$$\begin{tikzcd}
 & \mathrm{W}^\delta\text{-}\mathcal{O} \arrow[dr, dashed, "{\beta := (k\delta,\dots,k\delta)} \atop {[\leftarrow\ =\ \le]}"] & & & \\
\mathrm{E}^\alpha\text{-}\mathcal{O} \arrow[ur, "\delta := \max\{\alpha_1,\dots,\alpha_k\}"] \arrow[rr, bend left, "\beta := \alpha"] & & \mathrm{D}^\beta\text{-}\mathcal{O} \arrow[ll, bend left, "\alpha := \beta(1+\varepsilon)"] \arrow[rr, dashed, "{\sigma := (\beta_1^2,\beta_2)} \atop {[k\ =\ 2]}"] & & \mathrm{A}^\sigma\text{-}\mathcal{O} \\
 & & & \mathrm{S}^\gamma\text{-}\mathcal{O} \arrow[ul, "\beta := \gamma^2"'] \arrow[ur, "\sigma := \gamma"'] &
\end{tikzcd}$$

Figure 2: Implications between polynomial-time solvability of approximate solution notions for any k-objective problem $\mathcal{O}$ where $\varepsilon > 0$ can be chosen arbitrarily close to zero. Dashed lines indicate a conditional implication where the condition is shown in brackets. Note that $\alpha, \beta, \gamma, \sigma \in \mathbb{R}^k$, $\delta \in \mathbb{R}$ and W^δ-$\mathcal{O}$ is only defined if all objectives are minimized or all are maximized.

5.1 Notions of Multi-Objective Approximation

We discuss reasonable concepts of "approximately solving $\mathcal{O}$" for a k-objective problem $\mathcal{O} = (S, f, \leftarrow)$ where $\leftarrow$ is obtained from $\leftarrow_1, \dots, \leftarrow_k$. We start with the α-approximate version of E-$\mathcal{O}$, where $\alpha = (a_1, \dots, a_k)$ for $a_1, \dots, a_k \ge 1$.

E^α-$\mathcal{O}$ α-Approximate every-solution notion

Compute a set of solutions that α-dominates every solution.
Input: instance x
Output: some $S' \subseteq S^x$ such that $\forall s \in S^x\, \exists s' \in S' [s' \overset{\alpha}{\leftarrow} s]$

In Section 3 we argued that E-$\mathcal{O}$ is not an appropriate solution notion due to its large outputs. In contrast, approximations of E-$\mathcal{O}$ are very useful and feasible. Papadimitriou and Yannakakis [PY00] show that E^α-$\mathcal{O}$ has polynomial-size outputs for every $\alpha = (1+\varepsilon, \dots, 1+\varepsilon)$ with $\varepsilon > 0$. This means that there exists a polynomial p such that for all x there exists an $S' \subseteq S^x$ with $\#S' \le p(|x|)$ such that $\forall s \in S^x\, \exists s' \in S'[s' \overset{\alpha}{\leftarrow} s]$. Hence there exist small and quite precise approximations for S^x, but in many cases it is hard to compute these approximations. We also consider the following α-approximations.

A^{α}**-$\mathcal{O}$ α-Approximate arbitrary-optimum notion**

Compute a solution that weakly α-dominates an arbitrary optimal solution.

Input: instance x

Output: an $s \in S^x$ such that $s \overset{\alpha}{\leftarrow} t$ for some $t \in S^x_{\text{opt}}$ or report that $S^x = \emptyset$

S^{α}**-$\mathcal{O}$ α-Approximate specific-optimum notion**

Compute a solution that weakly α-dominates an optimal solution specified by a given cost vector.

Input: instance x, cost vector $c \in \mathbb{N}^k$

Output: an $s \in S^x$ such that $s \overset{\alpha}{\leftarrow} t \overset{\alpha}{\leftarrow} c$ for some $t \in S^x_{\text{opt}}$ or report that there is no $s \in S^x$ such that $s \leftarrow c$

D^{α}**-$\mathcal{O}$ α-Approximate dominating-solution notion**

Compute a solution that weakly α-dominates a given cost vector.

Input: instance x, cost vector $c \in \mathbb{N}^k$

Output: some $s \in S^x$ such that $s \overset{\alpha}{\leftarrow} c$ or report that there is no $s \in S^x$ such that $s \leftarrow c$

The performance of approximations that correspond to single-objective problems like W-$\mathcal{O}$ is specified by a real number δ instead of a vector of real numbers α. Here we consider the following δ-approximations, where $\delta \geq 1$. Similar to W-$\mathcal{O}$, we consider W^{δ}-$\mathcal{O}$ only for multi-objective problems where all objectives have to be minimized (resp., maximized). Note that in this case, $\leftarrow_1$ below can be replaced by any of the $\leftarrow_i$.

W^{δ}**-$\mathcal{O}$ δ-Approximate weighted-sum notion** (if all objectives are minimized or all maximized)

Single-objective problem that weights the objectives in a given way.

Input: instance x, weight vector $\omega \in \mathbb{N}^k$

Output: some $s \in S^x$ such that $\sum_{i=1}^{k} \omega_i f_i^x(s) \overset{\delta}{\leftarrow}_1 \sum_{i=1}^{k} \omega_i f_i^x(s')$ for all $s' \in S^x$ or report that $S^x = \emptyset$

$\mathrm{C}_{\mathrm{i}}^{\delta}$**-$\mathcal{O}$ δ-Approximate constraint notion for the i-th objective**

Single-objective problem that approximates the i-th objective while respecting constraints on the remaining objectives.

Input: instance x, constraints $b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_k \in \mathbb{N}$

Output: for $S^x_{\text{con}} = \{s \in S^x \mid f_j^x(s) \leftarrow_j b_j \text{ for all } j \neq i\}$ return an $s \in S^x_{\text{con}}$ with $s \overset{\delta}{\leftarrow}_i \operatorname{opt}_i(S^x_{\text{con}})$ or report that $S^x_{\text{con}} = \emptyset$

We disregard approximations for the lexicographical problem L-$\mathcal{O}$, since here it is not clear how to measure the performance of the approximation. Observe that each of these above notions coincides with its exact version if $\alpha = (1, \ldots, 1)$ or $\delta = 1$, respectively. Note that if $\mathcal{O}$ is a single-objective problem, then A^{δ}-$\mathcal{O}$, S^{δ}-$\mathcal{O}$, D^{δ}-$\mathcal{O}$, W^{δ}-$\mathcal{O}$, and C_1^{δ}-$\mathcal{O}$ are polynomial-time Turing equivalent to computing a δ-approximation for the single-objective problem.

Further research: Although D-$\mathcal{O}$ and $\mathrm{C_i}$-$\mathcal{O}$ are polynomial-time Turing equivalent, they can considerably differ with respect to their approximability. For example, for $\mathcal{O}$ = 2-EDC (2-objective exact disk cover) and for every $\delta > 1$ it holds that $\mathrm{E}^{(\delta,\delta)}$-$\mathcal{O}$ and $\mathrm{D}^{(\delta,\delta)}$-$\mathcal{O}$ are polynomial-time solvable. In contrast, for every $\delta > 1$ it holds that $\mathrm{C}_1^{(\delta,\delta)}$-$\mathcal{O}$ and $\mathrm{C}_2^{(\delta,\delta)}$-$\mathcal{O}$ are not polynomial-time solvable, unless P = NP [GRS08]. Regarding the nonapproximability of multi-objective problems only very few results are known. Here many questions remain open for further reserach.

Vassilvitskii and Yannakakis [VY05] investigate the problem of computing a good approximation for S^x_{opt} using as few solutions as possible. Here for a given instance x and a maximum number of solutions l, one has to find some $S \subseteq S^x$ with $\#S \leq l$ such that (1) there is some $\alpha \geq 1$ such that S is a solution to E^{α}-$\mathcal{O}$ on input x, and (2) there are no $\alpha' < \alpha$ and $S' \subseteq S^x$ with $\#S' \leq l$ such that S' is a solution to $\mathrm{E}^{\alpha'}$-$\mathcal{O}$ on input x.

For k-objective problems the performance of an approximation is given by a vector $\alpha \in \mathbb{R}^k$. Two such vectors can be incomparable and hence trade-offs are possible at the level of approximability. A typical 2-objective problem does not have *a single best* approximation ratio α, but there may exist a trade-off curve of incomparable best approximation ratios. Glaßer, Reitwießner, and Witek [GRW09] discuss evidence suggesting that metric 2-TSP has such approximation trade-offs.

5.2 Relations Between the Approximation Notions

We study relationships among the approximate problem notions defined above. Papadimitriou and Yannakakis [PY00] demonstrate a close connection between E-$\mathcal{O}$ and D-$\mathcal{O}$ in the sense that up to a factor of $1+\varepsilon$ both notions have the same approximability. We restate this result in a slightly extended version.

Theorem 5.1. *Let $\mathcal{O} = (S, f, \leftarrow)$ be a k-objective problem, $\alpha = (a_1, \ldots, a_k)$ where $a_i \geq 1$, and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_k)$ where $\varepsilon_i > 0$.*

1. E^{α}-$\mathcal{O}$ *polynomial-time solvable* $\Longrightarrow$ D^{α}-$\mathcal{O}$ *polynomial-time solvable*
2. D^{α}-$\mathcal{O}$ *polynomial-time solvable* $\Longrightarrow$ $\mathrm{E}^{\alpha(1+\varepsilon)}$-$\mathcal{O}$ *polynomial-time solvable (running time polynomially bounded in $|x| + \sum_i 1/\varepsilon_i$)*

Proof. Let $\mathcal{O} = (S, f, \leftarrow)$ be some k-objective problem.

1. Suppose we obtain S' as a solution to E^{α}-$\mathcal{O}$ in polynomial time. To solve D^{α}-$\mathcal{O}$ for some $c \in \mathbb{N}^k$ in polynomial time, we check whether there is some $s' \in S'$ with $s' \overset{\alpha}{\leftarrow} c$. If there is such $s' \in S'$, we are done. On the other hand, if there is no $s' \in S'$ with $s' \overset{\alpha}{\leftarrow} c$, there cannot be any $s \in S^x$ with $s \leftarrow c$ (otherwise S' is not a valid solution to E^{α}-$\mathcal{O}$, because there is no solution $s' \in S'$ that α-dominates s), and we are done as well.

2. We will construct some polynomial-sized lattice that covers the entire solution value space and then show how to find approximations of arbitrary solutions.

   By the definition of k-objective problems, there is some polynomial p such that $f^x_i(s) \leq 2^{p(|x|)}$ for every instance x, every solution $s \in S^x$, and $1 \leq i \leq k$.

   Fix some problem instance x, and let $\delta = \min\limits_{1 \leq i \leq k} \varepsilon_i$, $r = \lceil \frac{1}{\delta} \rceil$, and $t = r \cdot p(|x|) + 1$. We show that $(1+\delta)^t$ bounds the value of every objective. From

$$(1+\delta)^r = \sum_{j=0}^{r} \binom{r}{j} \delta^j = \binom{r}{0}\delta^0 + \binom{r}{1}\delta^1 + \sum_{j=2}^{r} \binom{r}{j}\delta^j \geq 1 + r \cdot \delta \geq 1 + \frac{1}{\delta} \cdot \delta = 2$$

   we obtain

$$f^x_i(s) \leq 2^{p(|x|)} < 2^{p(|x|)}(1+\delta) \leq ((1+\delta)^r)^{p(|x|)}(1+\delta) = (1+\delta)^{r \cdot p(|x|)+1} = (1+\delta)^t.$$

Next, consider the sets $I = \{(1+\delta)^j \mid 0 \leq j \leq t\}$ and $L = (I \cup \{0\})^k$. The elements of L cover the entire solution value space in the sense that for every solution $s \in S^x$ there is some $l \in L$ such that $l \overset{(1+\delta)}{\longleftarrow} s \leftarrow l$. It even holds that $\lfloor l \rceil \overset{(1+\delta)}{\longleftarrow} s \leftarrow \lfloor l \rceil$, where $\lfloor l \rceil$ denotes the vector obtained from l by rounding each component towards its direction of optimization. By $s \leftarrow \lfloor l \rceil$, given x and $\lfloor l \rceil$ as input, any solution for D^α-$\mathcal{O}$ must consist of some $s' \in S^x$ with $s' \overset{\alpha}{\leftarrow} \lfloor l \rceil$, thus $s' \overset{\alpha(1+\delta)}{\longleftarrow} s$.

To solve $\mathrm{E}^{\alpha(1+\varepsilon)}$-$\mathcal{O}$, it hence suffices to solve D^α-$\mathcal{O}$ for every $\lfloor l \rceil$ where $l \in L$ (note that $\delta \leq \varepsilon_i$ for all i). This is possible in time polynomial in $|x| + \sum_{i=1}^{k} \frac{1}{\varepsilon_i}$, since $\#L = (\lceil \frac{1}{\delta} \rceil \cdot p(|x|) + 3)^k$ and since D^α-$\mathcal{O}$ is solvable in time polynomial in $|x|$. □

Note that the algorithm for $\mathrm{E}^{\alpha(1+\varepsilon)}$-$\mathcal{O}$ calls D^α-$\mathcal{O}$ at every point of some polynomial-sized lattice built over ε. If, however, $\mathcal{O}$ is polynomially bounded, we can instead call D^α-$\mathcal{O}$ for every possible solution value and thereby obtain a solution to E^α-$\mathcal{O}$. Hence, for multi-objective problems that are polynomially bounded, E^α-$\mathcal{O}$ and D^α-$\mathcal{O}$ are equivalent.

Theorem 5.2. *Let $\mathcal{O} = (S, f, \leftarrow)$ be a k-objective problem and $\alpha = (a_1, \ldots, a_k)$ with $a_i \geq 1$.*

1. D^{α^2}-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{S}^\alpha$-$\mathcal{O}$ *and* A^α-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{S}^\alpha$-$\mathcal{O}$
2. $\mathrm{A}^{(\alpha_1^2, \alpha_2)}$-$\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{D}^\alpha$-$\mathcal{O}$ *if* $k = 2$
3. *If all objectives have to be minimized (resp., maximized) and* E^α-$\mathcal{O}$ *is polynomial-time solvable, then* $\mathrm{W}^{\max_i(\alpha_i)}$-$\mathcal{O}$ *is polynomial-time solvable.*

Proof. Let $\mathcal{O} = (S, f, \leftarrow)$ be a k-objective problem with $\leftarrow = (\leftarrow_1, \ldots, \leftarrow_k)$, instance x and $\alpha = (a_1, \ldots, a_k)$ with $a_i \geq 1$.

1. We call S^α-$\mathcal{O}$ for instance x and cost vector c as they are given in D^α-$\mathcal{O}$. If S^α-$\mathcal{O}$ reports that there is no $s \in S^x$ such that $s \leftarrow c$ we are done. Otherwise, S^α-$\mathcal{O}$ returns some $s \in S^x$, hence there is some $t \in S^x_{\mathrm{opt}}$ with $s \overset{\alpha}{\leftarrow} t \overset{\alpha}{\leftarrow} c$, which implies $s \overset{\alpha^2}{\leftarrow} c$, hence s is a solution to D^{α^2}-$\mathcal{O}$.

   For the second part, we call S^α-$\mathcal{O}$ with some cost vector c such that c is dominated by the entire solution value space. If $S^x \neq \emptyset$, we get some $s \in S^x$ such that $s \overset{\alpha}{\leftarrow} t \overset{\alpha}{\leftarrow} c$ for some $t \in S^x_{\mathrm{opt}}$, which solves A^α-$\mathcal{O}$.

2. Suppose $k = 2$, $S^x \neq \emptyset$, and let $s \in \mathrm{opt}_2(\mathrm{opt}_1(S^x))$. Clearly, s is optimal. Given polynomial-time solvability of D^α-$\mathcal{O}$, we perform a binary search over S^x that optimizes f_1^x and obtain a solution $s' \in S^x$ with $f_1^x(s') \overset{\alpha_1}{\leftarrow}_1 f_1^x(s)$. However, s' might have an inappropriate value in f_2^x. For that reason, we minimize f_2^x through a second binary search where we call D^α-$\mathcal{O}$ again and keep the first component of the cost vector fixed to $f_1^x(s')$. Since $f^x(s) \leftarrow (f_1^x(s'), f_2^x(s))$, this binary search finds a solution $\hat{s} \in S^x$ with $\hat{s} \overset{\alpha}{\leftarrow} (f_1^x(s'), f_2^x(s))$, and together with $f_1^x(s') \overset{\alpha_1}{\leftarrow}_1 f_1^x(s)$ we get $f_1^x(\hat{s}) \overset{\alpha_1^2}{\leftarrow}_1 f_1^x(s)$ and $f_2^x(\hat{s}) \overset{\alpha_2}{\leftarrow}_2 f_2^x(s)$.

3. Suppose all objectives have to be minimized (the theorem can be shown analogously if all objectives have to be maximized). For any instance x and weight vector $\omega = (\omega_1, \omega_2, \ldots, \omega_k)$, if $S^x \neq \emptyset$ then there is some $\hat{s} \in S^x$ that minimizes $\sum_{i=1}^{k} \omega_i f_i^x$. Let S' be a solution of E^α-$\mathcal{O}$. Then there must be a solution $s \in S'$ such that $s \overset{\alpha}{\leftarrow} \hat{s}$, hence

$f_i^x(s) \le \alpha_i f_i^x(\hat{s})$ for all i, which implies

$$\sum_{i=1}^k \omega_i f_i^x(s) \le \sum_{i=1}^k \omega_i \alpha_i f_i^x(\hat{s}) \le \max_i(\alpha_i) \sum_{i=1}^k \omega_i f_i^x(\hat{s}) \le \max_i(\alpha_i) \sum_{i=1}^k \omega_i f_i^x(s')$$

for all $s' \in S^x$. It hence suffices to return a solution $s^* \in S'$ that minimizes $\sum_{i=1}^k \omega_i f_i^x$, which can be extracted from S' in polynomial time, because S' has polynomial cardinality. □

Further research: Even though A-$\mathcal{O}$ can be reduced to all exact solution notions, it is not clear whether similar implications hold for A$^\alpha$-$\mathcal{O}$. For $k > 2$, the reducibility of A$^\alpha$-$\mathcal{O}$ to approximate problem notions other than S$^\alpha$-$\mathcal{O}$ remains open. Moreover, we are interested in improvements or lower bounds for the implications in Figure 2.

5.3 Pareto Minimization versus Scalar Minimization

We show that for multi-objective problems $\mathcal{O}$ where all objectives have to be minimized, approximability results translate from single-objective to multi-objective optimization. In particular, if W$^\delta$-$\mathcal{O}$ is polynomial-time solvable, then there exists some $c \ge 1$ such that D$^{c\delta}$-$\mathcal{O}$ and E$^{c\delta}$-$\mathcal{O}$ are polynomial-time solvable. Table 2 shows examples for such translations.

Let us first review some properties of norms as they are important in this section. A norm $||\cdot||$ on $\mathbb{R}^k$ is *monotone*, if for all vectors $x = (x_1, \dots, x_k)^T, y = (y_1, \dots, y_k)^T \in \mathbb{R}^k$ it holds that

$$|x_1| \le |y_1| \wedge \dots \wedge |x_k| \le |y_k| \quad \Rightarrow \quad ||x|| \le ||v||.$$

Two norms $||\cdot||_a$ and $||\cdot||_b$ on the same space are *equivalent*, if there exist constants $c_1, c_2 > 0$ such that for all x,

$$c_1||x||_b \le ||x||_a \le c_2||x||_b.$$

It is well known that all norms on $\mathbb{R}^k$ are equivalent. Important examples of norms on $\mathbb{R}^k$ are the *p-norms* $||(x_1, \dots, x_k)^T||_p = (\sum_{i=1}^k |x_i|^p)^{1/p}$ defined for real numbers $p \ge 1$ and the *maximum norm* $||(x_1, \dots, x_k)^T||_\infty = \max_i |x_i|$, which are all monotone, even if the components of the vectors are weighted by fixed non-negative numbers. Furthermore, for any $p \ge 1$ and any $x \in \mathbb{R}^k$ it holds that $||x||_\infty \le ||x||_p \le k^{1/p}||x||_\infty$.

The next two lemmas tell us how to translate approximations for weighted norms of vectors to the weak approximate dominance relation and vice-versa.

Lemma 5.3. *Let $k \ge 1$ and let $\leftarrow = (\le, \dots, \le)$ be the k-dimensional $\le$. For any norm $||\cdot||$ on $\mathbb{R}^k$ there is some $\hat{c} \ge 1$ such that for any $\delta \ge 1$ and $x, v \in \mathbb{N}^k$*

$$x \overset{(\delta,\dots,\delta)}{\leftarrow} v \qquad \Longrightarrow \qquad ||x|| \overset{\hat{c}\delta}{\le} ||v||.$$

In particular, if $||\cdot||$ is monotone then $\hat{c} = 1$, which is the case for any (weighted) p-norm and the (weighted) maximum norm.

Proof. First, suppose $||\cdot||$ is monotone. We get

$$\begin{aligned} x \overset{(\delta,\dots,\delta)}{\leftarrow} v &\Longrightarrow (x_1 \le \delta v_1 \wedge \dots \wedge x_k \le \delta v_k) \\ &\Longrightarrow ||x|| \le ||(\delta v_1, \dots, \delta v_k)^T|| \\ &\Longrightarrow ||x|| \le \delta||v||, \end{aligned}$$

which shows the lemma for the monotone case. Next, suppose $||\cdot||$ is not monotone. By the equivalence of norms on $\mathbb{R}^k$ there are constants $c_1, c_2 > 0$ such that $x \overset{(\delta,\dots,\delta)}{\leftarrow} v$ implies

$$c_1||x|| \leq ||x||_\infty \leq \delta||v||_\infty \leq c_2\delta||v||,$$

which yields the desired result for the general case with $\hat{c} = \frac{c_2}{c_1}$. Furthermore, observe that $c_1||x|| \leq ||x||_\infty \leq c_2||x||$ immediately implies $\frac{c_2}{c_1} \geq 1$. □

Lemma 5.4. *Let $k \geq 1$ and let $\leftarrow = (\leq, \dots, \leq)$ be the k-dimensional $\leq$. For any norm $||\cdot||$ on $\mathbb{R}^k$ there is some $\tilde{c} \geq 1$ such that for all $\delta \geq 1$ and $x, v \in \mathbb{N}^k$*

$$||Wx|| \leq \delta||Wv|| \qquad \Longrightarrow \qquad x \overset{(\tilde{c}\delta,\dots,\tilde{c}\delta)}{\leftarrow} v$$

where $W = \operatorname{diag}(\omega_1, \dots, \omega_k)$ and

$$\omega_i = \begin{cases} \lceil \delta\tilde{c} \rceil + 1 & \text{if } v_i = 0 \\ 1/v_i & \text{if } v_i \neq 0. \end{cases}$$

In particular, if $||\cdot||$ is a p-norm then $\tilde{c} = k^{1/p}$ and $\tilde{c} = 1$ for the maximum norm.

Proof. By the equivalence of norms on $\mathbb{R}^k$, there is some $C > 0$ such that $||x||_\infty \leq C||x||$ for all $x \in \mathbb{R}^k$. Set $\tilde{c} = C \max_{b\in\{0,1\}^k} ||b||$ ($b = (1, \dots, 1)^T$ if the norm is monotone) and observe that the following inequality holds for any i, any $x \in \mathbb{N}^k$ and any $v \in \mathbb{N}^k$.

$$\omega_i x_i = (Wx)_i \leq ||Wx||_\infty \leq C||Wx|| \leq C\delta||Wv|| \overset{(*)}{\leq} C\delta \max_{b\in\{0,1\}^k} ||b|| = \delta\tilde{c}$$

Note that for the inequality $(*)$, we used that $Wv \in \{0,1\}^k$.

If $v_i \neq 0$, this means that $x_i \leq \tilde{c}\delta v_i$ and for $v_i = 0$, we get $x_i \leq \frac{\delta\tilde{c}}{\lceil\delta\tilde{c}\rceil+1} < 1$ and thus $x_i = 0$ (since $x_i \in \mathbb{N}$), which again yields $0 = x_i \leq \tilde{c}\delta v_i = 0$.

Thus, we get $x_i \leq \tilde{c}\delta v_i$ for any i and the main part of the lemma is proved.

The second part is obtained by observing that $C = 1$ for any p-norm and for the maximum norm and that $||(1, \dots, 1)^T||_p = k^{1/p}$ for any p-norm and $||(1, \dots, 1)^T||_\infty = 1$. □

In order to apply these results to multiobjective optimization, we need to generalize the definition of the weighted-sum notion to a weighted-norm notion. For some k-objective problem $\mathcal{O}$ where all objectives have to be minimized, some norm $||\cdot||$ on $\mathbb{R}^k$, and some $\delta \geq 1$ we define the following.

$\mathrm{W}^\delta_{||\cdot||}$-$\mathcal{O}$ **δ-Approximate weighted-norm notion**

Single-objective problem that first weights the objectives and then applys a norm.

Input: instance x, weight vector $\omega \in \mathbb{N}^k$

Output: some $s \in S^x$ such that $||Wf^x(s)^T|| \leq \delta||Wf^x(s')^T||$ for all $s' \in S^x$ where $W = \operatorname{diag}(\omega_1, \dots, \omega_k)$ or report that $S^x = \emptyset$

This notion generalizes W^δ-$\mathcal{O}$, since W^δ-$\mathcal{O} = \mathrm{W}^\delta_{||\cdot||_1}$-$\mathcal{O}$, the δ-approximate weighted-1-norm notion. Note that the above definition can easily be extended for problems where all objectives have to be maximized.

Proposition 5.5. *For any norm $||\cdot||$ on $\mathbb{R}^k$ there is some $c \geq 1$ such that for any k-objective problem $\mathcal{O} = (S, f, \leq)$ and any $\delta \geq 1$ it holds that*

$$\mathrm{D}^{(\mathrm{c}\delta,\ldots,\mathrm{c}\delta)}\text{-}\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{W}^{\delta}_{||\cdot||}\text{-}\mathcal{O}.$$

In particular, if $||\cdot||$ is a p-norm then $c = k^{1/p}$ and $c = 1$ for the maximum norm.

Proof. We show how $\mathrm{D}^{(\mathrm{c}\delta,\ldots,\mathrm{c}\delta)}\text{-}\mathcal{O}$ can be solved in polynomial time relative to $\mathrm{W}^{\delta}_{||\cdot||}\text{-}\mathcal{O}$. Let the instance $x \in \mathbb{N}$ and the cost vector $v \in \mathbb{N}^k$ be the input. For the sake of clarity, we use $\leftarrow$ instead of $\leq$ in a multidimensional context. Let $\hat{c}, \tilde{c} \geq 1$ be the constants from Lemmas 5.3 and 5.4 corresponding to $||\cdot||$ and let $c = \hat{c}\tilde{c}$. Furthermore, let (as in Lemma 5.4) $\omega_i = \lceil \delta\tilde{c} \rceil + 1$ if $v_i = 0$ and $\omega_i = {}^1\!/_{v_i}$ if $v_i \neq 0$ and let V be the product of all nonzero entries in v ($V = 1$ if $v = (0,\ldots,0)$) and $\omega'_i = V\omega_i$. Note that the weights ω'_i are natural numbers, so we can call the algorithm for $\mathrm{W}^{\delta}_{||\cdot||}\text{-}\mathcal{O}$ with weights ω'_i. If we get a solution s from this call, we return s if $f^x(s) \overset{(c\delta,\ldots,c\delta)}{\leftarrow} v$. In all other cases, we report that there is no $s \in S^x$ such that $s \leftarrow v$.

For the correctness of this algorithm, we show that if there is some $p \in S^x$ with $p \leftarrow v$, then the algorithm returns some solution $s \in S^x$ such that $s \overset{(c\delta,\ldots,c\delta)}{\leftarrow} v$. Let $W = \mathrm{diag}(\omega_1,\ldots,\omega_k)$ and $W' = \mathrm{diag}(\omega'_1,\ldots,\omega'_k)$. Observe that any algorithm for $\mathrm{W}^{\delta}_{||\cdot||}\text{-}\mathcal{O}$ must return a solution if $S^x \neq \emptyset$, which is the case here. So the algorithm must return a solution $s \in S^x$ with $||W'f^x(s)^T|| \leq \delta||W'f^x(s')^T||$ for all $s' \in S^x$. In particular, from Lemma 5.3 we obtain

$$||W'f^x(s)^T|| \leq \delta||W'f^x(p)^T|| \leq \hat{c}\delta||W'v^T||.$$

Together with $W' = VW$ this yields

$$V||Wf^x(s)^T|| \leq V\hat{c}\delta||Wv^T||.$$

Now Lemma 5.4 tells us that $f^x(s) \overset{(c\delta,\ldots,c\delta)}{\leftarrow} v$ and so s is returned correctly.

For the runtime of the algorithm note that the size of the weights is polynomial in the size of v and the test if $f^x(s) \overset{(c\delta,\ldots,c\delta)}{\leftarrow} v$ is also possible in polynomial time.

Furthermore, the particular values for c result from the particular values for $\hat{c}$ and $\tilde{c}$. □

Corollary 5.6. *For any k-objective problem $\mathcal{O} = (S, f, \leq)$ and any $\delta \geq 1$ it holds that*

$$\mathrm{D}^{(\mathrm{k}\delta,\ldots,\mathrm{k}\delta)}\text{-}\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{W}^{\delta}\text{-}\mathcal{O}.$$

Proposition 5.7. *For any norm $||\cdot||$ on $\mathbb{R}^k$ there is some $c' \geq 1$ such that for any k-objective problem $\mathcal{O} = (S, f, \leq)$ and any $\delta \geq 1$ it holds that*

$$\mathrm{W}^{\mathrm{c}'\delta}_{||\cdot||}\text{-}\mathcal{O} \leq_{\mathrm{T}}^{\mathrm{p}} \mathrm{D}^{(\delta,\ldots,\delta)}\text{-}\mathcal{O}.$$

In particular, if $||\cdot||$ is a p-norm then $c' = k^{1/p}$ and $c' = 1$ for the maximum norm.

Proof. Let the instance x and the weights $\omega_1,\ldots,\omega_k \in \mathbb{N}$ be inputs for $\mathrm{W}^{\delta}_{||\cdot||}\text{-}\mathcal{O}$. For the sake of clarity, we use $\leftarrow$ instead of $\leq$ in multidimensional contexts. Let $\omega'_i = {}^1\!/_{\omega_i}$ if $\omega_i \neq 0$ and otherwise, let ω'_i be some number larger than any possible output of f^x. Using binary search with queries to $\mathrm{D}^{(\delta,\ldots,\delta)}\text{-}\mathcal{O}$ we determine some solution $s \in S^x$ and some $r \in \mathbb{N}$ such that $s \overset{(\delta,\ldots,\delta)}{\leftarrow} (r\omega'_1,\ldots,r\omega'_k) \in \mathbb{N}^k$ and there is no $s' \in S^x$ such that $s' \leftarrow (r\omega'_1 - 1,\ldots,r\omega'_k - 1)$. This means that for any $s' \in S^x$, there is some i such that $f^x_i(s') \geq r\omega'_i$. This can only happen if $\omega_i \neq 0$ and thus we get $||(\omega_1 f^x_1(s'),\ldots,\omega_k f^x_k(s'))^T||_\infty \geq r$ for all $s' \in S^x$.

Regarding s, we get $\omega_i f_i^x(s) \leq \delta r$ for all i, which means that $||(\omega_1 f_1^x(s), \dots, \omega_k f_k^x(s))^T||_\infty \leq \delta r \leq \delta||(\omega_1 f_1^x(s'), \dots, \omega_k f_k^x(s'))^T||_\infty$ for all $s' \in S^x$ and thus s is a correct output for the problem if the norm is the maximum norm. Otherwise, there are constants $c_1, c_2 > 0$ by the equivalence of norms such that $||(\omega_1 f_1^x(s), \dots, \omega_k f_k^x(s))^T|| \leq \frac{c_2}{c_1}\delta||(\omega_1 f_1^x(s'), \dots, \omega_k f_k^x(s'))^T||$ for all $s' \in S^x$, which completes the first part of the assertion with $c' = \frac{c_2}{c_1}$.

The second part is obtained by observing that $\frac{c_2}{c_1} = k^{1/p}$ for a p-norm. □

Corollary 5.8. *For any k-objective problem $\mathcal{O} = (S, f, \leq)$ and any $\delta \geq 1$ it holds that*

$$\mathrm{D}^{(\delta,\dots,\delta)}\text{-}\mathcal{O} \equiv_{\mathrm{T}}^{\mathrm{p}} \mathrm{W}^{\delta}_{||\cdot||_\infty}\text{-}\mathcal{O}.$$

Note that the factor c' provided by Proposition 5.7 is quite large, especially for the 1-norm. By Theorems 5.2.3 and 5.1.2 we know that (for $\alpha_1 = \dots = \alpha_k$) the much better factor $(1+\varepsilon)$ is possible by first solving $\mathrm{E}^{\alpha(1+\varepsilon)}$-$\mathcal{O}$ for some $\varepsilon > 1$. This result can be extended to general monotone norms.

Proposition 5.9. *Let $\mathcal{O} = (S, f, \leftarrow)$ be a k-objective problem where $\leftarrow = \leq$ or $\leftarrow = \geq$, let $||\cdot||$ be some monotone norm on $\mathbb{R}^k$ and $\alpha = (a_1, \dots, a_k)$ with $a_i \geq 1$.*

If E^α-$\mathcal{O}$ is polynomial-time solvable, then $\mathrm{W}^{\mathrm{max_i(\alpha_i)}}_{||\cdot||}$-$\mathcal{O}$ is polynomial-time solvable.

Proof. We show this similarly to Theorem 5.2.3. Suppose all objectives have to be minimized (the proposition can be shown analogously if all objectives have to be maximized). For any instance x and weight vector $\omega = (\omega_1, \omega_2, \dots, \omega_k) \in \mathbb{N}^k$, if $S^x \neq \emptyset$ then there is some $\hat{s} \in S^x$ that minimizes $||W f_i^{x^T}||$ for $W = \mathrm{diag}(\omega_1, \dots, \omega_k)$. Let S' be a solution of E^α-$\mathcal{O}$. Then there must be some $s \in S'$ such that $s \overset{\alpha}{\leftarrow} \hat{s}$, hence

$$f_i^x(s) \leq \alpha_i f_i^x(\hat{s}) \leq \max_i(\alpha_i) f_i^x(\hat{s})$$

for all i, which implies

$$||Wf^x(s)^T|| \leq ||W \max_i(\alpha_i) f^x(\hat{s})^T|| \leq \max_i(\alpha_i)||Wf^x(\hat{s})^T|| \leq \max_i(\alpha_i)||Wf^x(s')^T||$$

for all $s' \in S^x$. It hence suffices to return a solution $s^* \in S'$ that minimizes $||Wf^{x^T}||$, which can be extracted from S' in polynomial time, because S' has polynomial cardinality. □

Corollary 5.10. *The following statements are equivalent for some k-objective problem $\mathcal{O} = (S, f, \leq)$:*

- D^α-$\mathcal{O}$ *is polynomial-time solvable for some $\alpha = (\alpha_1, \dots, \alpha_k)$ with $\alpha_i \geq 1$.*
- W^δ-$\mathcal{O}$ *is polynomial-time solvable for some $\delta \geq 1$.*
- $\mathrm{W}^\delta_{||\cdot||}$-$\mathcal{O}$ *is polynomial-time solvable for some norm $||\cdot||$ on $\mathbb{R}^k$ and some $\delta \geq 1$.*

Further research: By adjusting the individual weights in the reductions, one can change the approximation for D^α-$\mathcal{O}$ in the sense that one can improve the factor for some criteria at the expense of others. It can be further investigated if there are problems where this yields new approximation results, especially for problems where approximation trade-offs are assumed to exist.

Problem $\mathcal{O}$	Definition	solvable in P	Ref.
2-objective minimum vertex cover	$(S, f, \leq)$, instances are $\mathbb{N}^2$-node-labeled graphs $G = (V, E, l)$, $S^G = \{C \mid C$ is a vertex cover for $G\}$ and $f^G(C) = \sum_{v \in C} l(v)$	$\mathrm{D}^{(4,4)}$-$\mathcal{O}$	[BYE85] [MS85]
2-objective minimum TSP with repetitions	$(S, f, \leq)$, instances are $\mathbb{N}^2$-edge-labeled graphs $G = (V, E, l)$, $S^G = \{H \mid H \subseteq E$ is a closed walk visting each vertex at least once$\}$ and $f^G(H) = \sum_{e \in H} l(e)$	$\mathrm{D}^{(3,3)}$-$\mathcal{O}$	[Chr76]
2-objective minimum k-spanning tree	$(S, f, \leq)$, instances are pairs (G, k) where $G = (V, E, l)$ is an $\mathbb{N}^2$-edge-labeled graph, $k \leq \lvert V \rvert$, $S^G = \{T \mid T \subseteq E$ is a subtree of $G\}$, and $f^G(T) = \sum_{e \in T} l(e)$	$\mathrm{D}^{(6,6)}$-$\mathcal{O}$	[Gar96]
2-objective minimum k-cut	$(S, f, \leq)$, instances are pairs (G, k) where $G = (V, E, l)$ is an $\mathbb{N}^2$-edge-labeled graph, $2 \leq k \leq \lvert V \rvert$, $S^G = \{(C_1, \ldots, C_k) \mid (C_1, \ldots, C_k)$ is a partition of $V\}$, and $f^G(C_1, \ldots, C_k) = \sum l(e)$ where the sum runs over all edges $e = (u, v)$ such that u, v are in different C_i	$\mathrm{D}^{(4,4)}$-$\mathcal{O}$	[SV91]

Table 2: Examples for multi-objective approximability results directly obtained from known single-objective results by applying Corollary 5.6.

5.4 Pareto Maximization versus Scalar Maximization

The previous subsection showed that for problems where all objectives have to be minimized, approximability results translate from single-objective to multi-objective optimization (Corollary 5.6). We now show the limits of such translations and prove that they are impossible for maximization problems, unless P = NP. More precisely, the following holds for a restricted version $\mathcal{O}$ of the 2-objective maximum clique problem.

1. W-$\mathcal{O}$ is polynomial-time solvable.
2. For every $\alpha_1, \alpha_2 \geq 1$, $\mathrm{E}^{(\alpha_1,\alpha_2)}$-$\mathcal{O}$ and $\mathrm{D}^{(\alpha_1,\alpha_2)}$-$\mathcal{O}$ are not polynomial-time solvable, unless P = NP.

Definition 5.11 (***k***-Objective Maximum Clique)**.**

> ***k*-CLIQUE** $= (S, f, \geq)$ *where instances are* $\mathbb{N}^k$*-node-labeled graphs* $G = (V, E, l)$, $S^G = \{C \mid C \subseteq V \wedge C \times C \subseteq E\}$, *and* $f^G(C) = \sum_{v \in C} l(v)$.

We consider 2-CLIQUE restricted to instances that consist of an arbitrary graph $G = (V, E, l)$ with labels $(1, 1)$ and two additionally nodes x, y that have no connections to other nodes and that have labels $(2n + 1, 0)$ and $(0, 2n + 1)$, where $n = \#V$. More precisely, the set of restricted instances is defined as

$$\mathcal{R} = \{G \mid G = (V \cup \{x, y\}, E \cup \{(x, x), (y, y)\}, l) \text{ where } (V, E) \text{ is a graph with } n \text{ nodes, } l(x) = (2n+1, 0),\ l(y) = (0, 2n+1), \text{ and } l(v) = (1, 1) \text{ for } v \in V\}.$$

> **2-CLIQUE$_{\mathbf{restr}}$** = 2-CLIQUE restricted to instances from $\mathcal{R}$.

Proposition 5.12. *Let* $\mathcal{O} = \text{2-CLIQUE}_{\text{restr}}$.

1. W-$\mathcal{O}$ *is polynomial-time solvable.*
2. *There is no* $\alpha \in \mathbb{R}^2$ *such that* E^{α}-$\mathcal{O}$ *is polynomial-time solvable, unless* P = NP.
3. *There is no* $\alpha \in \mathbb{R}^2$ *such that* D^{α}-$\mathcal{O}$ *is polynomial-time solvable, unless* P = NP.

Proof. 1. On input $(V \cup \{x, y\}, E \cup \{(x, x), (y, y)\}, c) \in \mathcal{R}$ and $(w_1, w_2) \in \mathbb{N}^k$, the algorithm outputs $\{x\}$ if $w_1 \geq w_2$, and $\{y\}$ otherwise.

2. Assume that E^α-$\mathcal{O}$ is polynomial-time solvable for some $\alpha \in \mathbb{R}^2$. We may assume $\alpha = (c, c)$ for some $c \geq 1$. We show that CLIQUE is c-approximable which implies P = NP [ALM+92].

   Let $G = (V, E)$ be a graph, let $n = \#V$, and let m be the size of the maximal clique in G. Define the 2-CLIQUE$_{\mathrm{restr}}$ instance $G' = (V \cup \{x, y\}, E \cup \{(x, x), (y, y)\}, l)$ according to the definition of $\mathcal{R}$. Now consider the solution algorithm for E^α-$\mathcal{O}$ on input G'. Since $S_{\mathrm{opt}}^{G'}$ contains a solution with value (m, m), the output of the algorithm must contain some $S \subseteq V \cup \{x, y\}$ such that $c \cdot f(S) \geq (m, m)$. From $f_1(\{y\}) = 0$ and $f_2(\{x\}) = 0$ it follows that $S \neq \{y\}$ and $S \neq \{x\}$. Therefore, $S \subseteq V$, since x and y have no edges with other nodes. Hence $c \cdot |S| \geq m$, i. e., S is a clique of size m/c in G.

3. Follows from 2. and Theorem 5.1. □

The example 2-CLIQUE$_{\mathrm{restr}}$ shows the disadvantage of the weighted-sum notion for maximization problems. This effect does not only appear at artificially constructed multi-objective problems. For instance, consider the 2-objective maximum matching problem. Here W-$\mathcal{O}$ is polynomial-time solvable. Nevertheless, the following instance shows that the solutions for W-$\mathcal{O}$ are not good approximations for the set of non-dominated solutions.

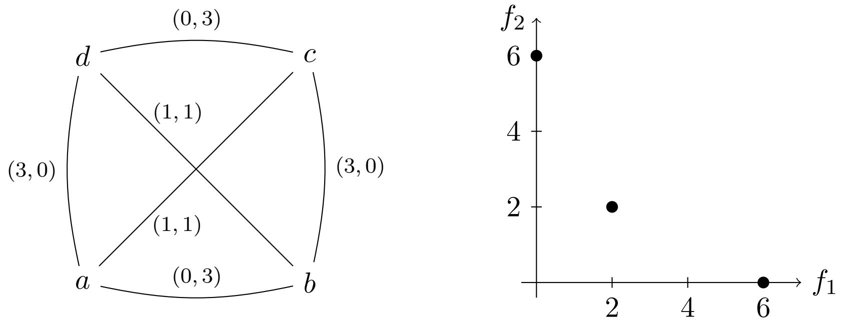

Figure 3: The right-hand side shows the values of the Pareto set of the 2-objective maximum matching instance that is shown on the left. There are exactly the three optimal values $(0, 6)$, $(6, 0)$, and $(2, 2)$, but the weighted-sum notion W-$\mathcal{O}$ finds only solutions with values $(0, 6)$ and $(6, 0)$. There is no $\alpha \in \mathbb{R}^2$ such that $(0, 6)$ or $(6, 0)$ weakly α-dominates $(2, 2)$.

References

[ALM+92] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the intractability of approximation problems. In *Proceedings 33rd Symposium on the Foundations of Computer Science*, pages 14–23. IEEE Computer Society Press, 1992.

[Bak74] K. R. Baker. *Introduction to Sequencing and Scheduling.* Wiley, New York, 1974.

[BLS84] R. V. Book, T. Long, and A. L. Selman. Quantitative relativizations of complexity classes. *SIAM Journal on Computing*, 13:461–487, 1984.

[BYE85] R. Bar-Yehuda and S. Even. A local ratio theorem for approximating the weighted vertex cover problem. In *Analysis and Design of Algorithms for Combinatorial Problems*, volume 25 of *Annals of Discrete Mathematics*, pages 27–46, 1985.

[Chr76] N. Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA, 1976.

[EG00] M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spectrum*, 22(4):425–460, 2000.

[EG02] M. Ehrgott and X. Gandibleux, editors. *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Survey*, volume 52 of *Kluwer's International Series in Operations Research and Management Science*. Kluwer Academic Publishers, 2002.

[Ehr05] M. Ehrgott. *Multicriteria Optimization*. Springer Verlag, 2005.

[Gar96] N. Garg. A 3-approximation for the minimum tree spanning k vertices. In *37th Annual Symposium on Foundations of Computer Science*, pages 302–309. IEEE Computer Society Press, 1996.

[GLLK79] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287–326, 1979.

[GRS08] C. Glaßer, C. Reitwießner, and H. Schmitz. Multiobjective disk cover admits a PTAS. In *Proceedings of 19th International Symposium on Algorithms and Computation*, volume 5369 of *Lecture Notes in Computer Science*. Springer Verlag, 2008.

[GRW09] C. Glaßer, C. Reitwießner, and M. Witek. Improved and derandomized approximations for two-criteria metric traveling salesman. Technical Report TR09-076, Electronic Colloquium on Computational Complexity, 2009.

[Han79] P. Hansen. Bicriterion path problems. In *Proceedings of the 3rd Conference on Multiple Criteria Decision Making Theory and Application*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, pages 109–127. Springer Verlag, 1979.

[Hoo92] J. A. Hoogeveen. *Single-Machine Bicriteria Scheduling*. PhD thesis, Technical University of Eindhoven, 1992.

[HvdV90] J. A. Hoogeveen and S. L. van de Velde. Polynomial-time algorithms for single-machine multicriteria scheduling. Technical Report BS-R9008, Centre for Mathematics and Computer Science, Amsterdam, 1990.

[LV93] C.-Y. Lee and G. L. Vairaktarakis. Complexity of single machine hierarchical scheduling: A survey. In P. M. Pardalos, editor, *Complexity in Numerical Optimization*, pages 269–298. World Scientific, 1993.

[MA78] K. L. Manders and L. M. Adleman. NP-complete decision problems for binary quadratics. *Journal of Computer and System Sciences*, 16(2):168–184, 1978.

[MS85] B. Monien and E. Speckenmeyer. Ramsey numbers and an approximation algorithm for the vertex cover problem. *Acta Informatica*, 22(1):115–123, 1985.

[PY82] C. H. Papadimitriou and M. Yannakakis. The complexity of restricted spanning tree problems. *J. ACM*, 29(2):285–309, 1982.

[PY00] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *FOCS '00: Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pages 86–95, Washington, DC, USA, 2000. IEEE Computer Society.

[Sel94] A. L. Selman. A taxonomy on complexity classes of functions. *Journal of Computer and System Sciences*, 48:357–381, 1994.

[SV91] H. Saran and V. V. Vazirani. Finding k-cuts within twice the optimal. In *32nd Annual Symposium on Foundations of Computer Science*, pages 743–751. IEEE Computer Society Press, 1991.

[VY05] S. Vassilvitskii and M. Yannakakis. Efficiently computing succinct trade-off curves. *Theoretical Computer Science*, 348(2-3):334–356, 2005.