Decision Making Based on Approximate and Smoothed Pareto Curves

Heiner Ackermann, Alantha Newman, Heiko Röglin *, Berthold Vöcking

RWTH Aachen, Lehrstuhl für Informatik I, D-52056 Aachen, Germany

---

Abstract

We consider bicriteria optimization problems and investigate the relationship between two standard approaches to solving them: (i) computing the *Pareto curve* and (ii) the so-called *decision maker's approach* in which both criteria are combined into a single (usually non-linear) objective function. Previous work by Papadimitriou and Yannakakis showed how to efficiently approximate the Pareto curve for problems like Shortest Path, Spanning Tree, and Perfect Matching. We wish to determine for which classes of combined objective functions the approximate Pareto curve also yields an approximate solution to the decision maker's problem. We show that an FPTAS for the Pareto curve also gives an FPTAS for the decision maker's problem if the combined objective function is growth bounded like a quasi-polynomial function. If the objective function, however, shows exponential growth then the decision maker's problem is NP-hard to approximate within any polynomial factor. In order to bypass these limitations of approximate decision making, we turn our attention to Pareto curves in the probabilistic framework of smoothed analysis. We show that in a smoothed model, we can efficiently generate the (complete and exact) Pareto curve with a small failure probability if there exists an algorithm for generating the Pareto curve whose worst case running time is pseudopolynomial. This way, we can solve the decision maker's problem w. r. t. any non-decreasing objective function for randomly perturbed instances of, e. g., Shortest Path, Spanning Tree, and Perfect Matching.

Key words: Decision Making, Multicriteria Optimization, Pareto Curve, Smoothed Analysis

---

* Corresponding author. Tel.: +49 241 80 21107; fax: +49 241 80 22216
Email addresses: ackermann@cs.rwth-aachen.de (Heiner Ackermann), alantha@cs.rwth-aachen.de (Alantha Newman), roeglin@cs.rwth-aachen.de (Heiko Röglin), voecking@cs.rwth-aachen.de (Berthold Vöcking).

1 Introduction

We study *bicriteria optimization* problems, in which there are two criteria, say cost and weight, that we are interested in optimizing. In particular, we consider bicriteria Spanning Tree, Shortest Path and Perfect Matching problems. For such problems with more than one objective, it is not immediately clear how to define an optimal solution. However, there are two common approaches to bicriteria optimization problems.

The first approach is to generate the set of *Pareto optimal* solutions, also known as the *Pareto set*. A solution S^* is Pareto optimal if there exists no other solution S that *dominates* S^*, i. e., has cost and weight less or equal to the cost and weight of S^* and at least one inequality is strict. The set of cost/weight combinations of the Pareto optimal solutions is called the *Pareto curve*. Often it is sufficient to know only one solution for each possible cost/weight combination. Thus, we assume that the Pareto set is reduced and does not contain two solutions with equal cost and equal weight. Under this assumption there is a one-to-one mapping between the elements in the reduced Pareto set and the points on the Pareto curve.

The second approach is to compute a solution that minimizes some *non-decreasing* function $f : \mathbb{R}_+^2 \to \mathbb{R}_+$. This approach is often used in the field of *decision making*, in which a decision maker is not interested in the whole Pareto set but in a single solution with certain properties. For example, given a graph $G = (V, E)$ with cost $c(e)$ and weight $w(e)$ on each edge, one could be interested in finding an s-t-path P that minimizes the value $(\sum_{e\in P} w(e))^2 + (\sum_{e\in P} c(e))^2$. For a given function $f : \mathbb{R}_+^2 \to \mathbb{R}_+$ and a bicriteria optimization problem Π we denote by f-Π the problem of minimizing f over all solutions of Π.

Note that these two approaches are actually related: for any *non-decreasing* function f, there is a solution that minimizes f that is also Pareto optimal. A function $f : \mathbb{R}_+^2 \to \mathbb{R}_+$ is non-decreasing if $f(x_1, y_1) \le f(x_2, y_2)$ for any $x_1, x_2, y_1, y_2 \in \mathbb{R}_+$ with $x_1 \le x_2$ and $y_1 \le y_2$. Thus, if for a particular bicriteria optimization problem, we can find the Pareto set efficiently and it has polynomial size, then we can efficiently find a solution that minimizes any given non-decreasing function. It is known, however, that there are instances of Spanning Tree, Shortest Path and Perfect Matching problems such that even the reduced Pareto set is exponentially large [6]. Moreover, while efficient (i. e. polynomial in the size of the Pareto set) algorithms are known for a few standard bicriteria optimization problems such as the Shortest Path problem [7,18], it is not known how to generate the Pareto set efficiently for other well-studied bicriteria optimization problems such as the Spanning Tree and the Perfect Matching problem.

There has been a long history of *approximating* the Pareto set starting with the pioneering work of Hansen [7] on the SHORTEST PATH problem. We say a solution S is ε-approximated by another solution S' if $c(S')/c(S) \leq 1+\varepsilon$ and $w(S')/w(S) \leq 1+\varepsilon$ where $c(S)$ and $w(S)$ denote the total cost and weight of a solution S. We say that $\mathcal{P}_\varepsilon$ is an ε-approximation of a Pareto set $\mathcal{P}$ if for any solution $S \in \mathcal{P}$ there is a solution $S' \in \mathcal{P}_\varepsilon$ that ε-approximates it. Papadimitriou and Yannakakis showed that for any Pareto set $\mathcal{P}$, there is an ε-approximation of $\mathcal{P}$ with polynomially many points [13] (w. r. t. the input size and $1/\varepsilon$). Furthermore they gave necessary and sufficient conditions under which there is an FPTAS to generate $\mathcal{P}_\varepsilon$. Vassilvitskii and Yannakakis [17] showed how to compute ε-approximate Pareto curves of almost minimal size.

1.1 Previous Work

There exists a vast body of literature that focuses on f-Π problems. For instance it is well known that, if f is a concave function, an optimal solution of the f-Π problem can be found on the border of the convex hull of the solutions [9]. For some problems there are algorithms generating this set of solutions. In particular, for the SPANNING TREE Problem it is known that there are only polynomially many solutions on the border of the convex hull [5], and efficient algorithms for enumerating them exist [1]. Thus, there are polynomial-time algorithms for solving f-SPANNING TREE if f is concave. Katoh has described how one can use f-SPANNING TREE problems with concave objective functions to solve many other problems in combinatorial optimization [10]. For instance, a well studied application is the MINIMUM COST RELIABILITY SPANNING TREE Problem, where one is interested in finding a spanning tree minimizing the ratio of cost to reliability. This approach, however, is limited to optimizing the ratio of these two criteria. It is also known how to solve the f-SHORTEST PATH problem for functions f being both pseudoconcave and pseudoconvex in polynomial time [8]. Tsaggouris and Zaroliagis [15] investigated the NON-ADDITIVE SHORTEST PATH Problem (NASP), which is to find a path P minimizing $f_c(c(P)) + f_w(w(P))$, for some convex functions f_c and f_w. This problem arises as core problem in different applications, e. g., in the context of computing traffic equilibria. They developed exact algorithms with exponential running time using a Lagrangian relaxation and the so called *Extended Hull Algorithm* to solve NASP.

We consider bicriteria optimization problems in the smoothed analysis framework of Spielman and Teng [14]. Spielman and Teng consider a semi-random input model where an adversary specifies an input which is then randomly perturbed. Input instances occurring in practice usually possess a certain structure but usually also have small random influences. Thus, one can hope that semi-random input models are more realistic than worst case and average case input models since the adversary can specify an arbitrary input with a certain structure that is subsequently only slightly perturbed. Since the seminal work of Spielman and Teng explaining the efficiency of the Simplex method in practical applications [14], many other problems have been considered in the framework of smoothed analysis. Of particular relevance to the results in this paper are the results of Beier and Vöcking [3,4]. First, they showed that the expected number of Pareto optimal solutions of any bicriteria optimization problem with two linear objective functions is polynomial if the coefficients in the objective functions are randomly perturbed [3]. Then they gave a complete characterization which linear binary optimization problems have polynomial smoothed complexity, namely they showed that a linear binary optimization problem has polynomial smoothed complexity if and only if there exists an algorithm whose worst case running time is pseudopolynomially bounded in the perturbed coefficients [4]. The only way to apply their framework to multicriteria optimization is by moving all but one of the criteria from the objective function to the constraints.

1.2 Our Results

We study the complexity of the bicriteria optimization problems f-Shortest Path, f-Spanning Tree and f-Perfect Matching under different classes of functions f. Our study begins with an analysis showing that these problems are NP-hard even under seemingly harmless objective functions of the form *Minimize* $(\sum_{e\in S} c(e))^a + (\sum_{e\in S} w(e))^b$, where a, b are arbitrary natural numbers with $a \geq 2$ or $b \geq 2$. Thus, we focus on the approximability of these problems. An FPTAS to approximate the Pareto curve of a problem Π can be transformed into an FPTAS for f-Π for any polynomial function f easily. We show that this transformation also works for *quasi-polynomial* functions and, more generally, for non-decreasing functions whose first derivative is bounded from above like the first derivative of a quasi-polynomial function. (A similar result has been shown recently in an independent work by Tsaggouris and Zaroliagis [16].) Additionally, we show that the restriction to quasi-polynomial growth is crucial.

In order to bypass the limitations of approximate decision making seen above, we turn our attention to Pareto curves in the probabilistic framework of smoothed analysis. We show that in a smoothed model, we can efficiently generate the (complete and exact) Pareto curve of Π with a small failure probability if there exists an algorithm for generating the Pareto curve whose worst case running time is pseudopolynomial (w. r. t. costs and weights). Previously, it was known that the number of Pareto optimal solutions is polynomially bounded if the input numbers are randomly perturbed [3]. This result, however, left open the question of how to generate the set of Pareto-optimal solutions efficiently (except for the Shortest Path problem). The key result in the smoothed analysis presented in this paper is that typically the smallest gap (in cost and weight) between neighboring solutions on the Pareto curve is bounded by $n^{-O(1)}$ from below. This result enables us to generate the complete Pareto curve by taking into account only a logarithmic number of bits of each input number. This way, an algorithm with pseudopolynomial worst-case complexity for generating the Pareto curve can be turned into an algorithm with polynomial smoothed complexity.

It can easily be seen that, for any bicriteria problem Π, a pseudopolynomial algorithm for the exact and single objective version of Π (e. g. an algorithm for answering the question "Does there exist a spanning tree with costs exactly C?") can be turned into an algorithm with pseudopolynomial worst-case complexity for generating the Pareto curve. Therefore, in the smoothed model, there exists a polynomial-time algorithm for enumerating the Pareto curve of Π with small failure probability if there exists a pseudopolynomial algorithm for the exact and single objective version of Π. Furthermore, given the exact Pareto curve for a problem Π, one can solve f-Π exactly. Thus, in our smoothed model, we can, for example, find spanning trees that minimize functions that are hard to approximate within any factor in the worst case.

2 Approximating Bicriteria Optimization Problems

In this section, we consider bicriteria optimization problems in which the goal is to minimize a single objective function that takes two criteria as inputs. We consider functions of the form $f(x, y)$ where x represents the total cost of a solution and y represents the total weight of a solution. In Section 2.1, we present NP-hardness and inapproximability results for the f-Spanning Tree, f-Shortest Path, and f-Perfect Matching problems for general classes of functions. In Section 2.2, we show that we can give an FPTAS for any f-Π problem for a large class of quasi-polynomially bounded non-decreasing functions f if there is an FPTAS for generating an ε-approximate Pareto curve for Π. Papadimitriou and Yannakakis showed how to construct such an FPTAS for approximating the Pareto curve of Π given an exact pseudopolynomial algorithm for the problem [13]. For the exact s-t-Path problem, dynamic programming yields a pseudopolynomial algorithm [18]. For the exact Spanning Tree problem, Barahona and Pulleyblank gave a pseudopolynomial algorithm [2]. For the exact Matching problem, there is a fully polynomial RNC scheme [12,11]. Thus, for any quasi-polynomially bounded non-decreasing objective function, these problems have an FPTAS.

In this section, we present NP-hardness results for the bicriteria f-Spanning Tree, f-Shortest Path, and f-Perfect Matching problems in which the goal is to find a feasible solution S that minimizes an objective function of the form $f(x, y) = x^a + y^b$, where $x = c(S)$, $y = w(S)$, and $a, b \in \mathbb{N}$ are constants with $a \geq 2$ or $b \geq 2$. The NP-hardness of such functions follows quite directly from a simple reduction from Partition when $a = b$. When a and b differ, one can modify this reduction slightly by scaling the weights.

Theorem 1 *Let $f(x, y) = x^a + y^b$ with $a, b \in \mathbb{N}$ and $a \geq 2$ or $b \geq 2$. Then the f-Spanning Tree, f-Shortest Path, and f-Perfect Matching problems are NP-hard.*

PROOF. By simple reductions from Partition, one can prove that it is NP-hard to decide whether a graph with edge costs and weights has a spanning tree (or s-t-path or perfect matching) with cost at most C and weight at most W, where $C, W \in \mathbb{R}$ [6]. We do not reproduce these reductions completely, but mention only their key properties and adapt them appropriately to prove the lemma.

A Partition instance consists of n natural numbers $a_1, \ldots, a_n \in \mathbb{N}$ and the goal is to decide whether there is a partition of these numbers into disjoint sets $\mathcal{A}_1$ and $\mathcal{A}_2$ such that $\sum_{a_i \in \mathcal{A}_1} a_i = \sum_{a_i \in \mathcal{A}_2} a_i = A/2$ with $A = \sum_{i=1}^{n} a_i$. The graphs constructed in the reductions contain for each number a_i, two edges e_i^1 and e_i^2 with cost a_i and weight 0 and with cost 0 and weight a_i, respectively. In each feasible solution, exactly one of these edges is contained for each number a_i. Furthermore, the graphs possess the property that for each partition $(\mathcal{A}_1, \mathcal{A}_2)$ there exists a feasible solution containing edge e_i^1 for every $i \in \mathcal{A}_1$ and edge e_i^2 for every $i \in \mathcal{A}_2$. Besides the edges e_i^1 and e_i^2, the graphs contain only edges with costs and weights 0. Graphs possessing these properties for the Spanning Tree, Shortest Path, and Perfect Matching problem are depicted in Figures 1 and 2.

Due to the aforementioned properties, every feasible solution S satisfies $c(S) + w(S) = A$. Note, that for every $a \geq 2$, under the conditions $x \geq 0$, $y \geq 0$, and $x + y = A$, the function $f(x, y) = x^a + y^a$ takes its unique minimum at $x = y = A/2$. Therefore, the reductions presented above show that for functions of this type the considered f-Π problems are NP-hard as by solving the f-Π problems one can decide whether the given numbers $a_1, \ldots, a_n$ can be partitioned into sets of equal size.

Now we modify the reductions presented above slightly to show that the considered f-Π problems are also NP-hard for functions $f(x, y) = x^a + y^b$ with

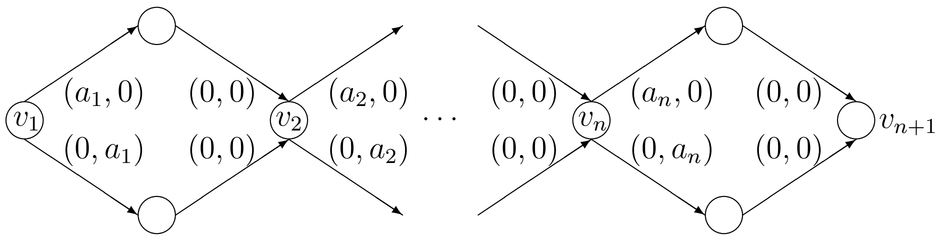

Fig. 1. The graph constructed in the reductions from Partition to the s-t-Path problem and the Spanning Tree problem. In the reduction to the Spanning Tree problem the edges are undirected.

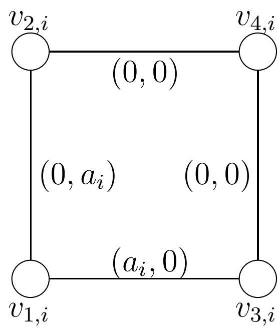

Fig. 2. For each number a_i, the graph constructed in the reduction from Partition to the Perfect Matching problem contains one of these gadgets.

$a \neq b$. We use the reductions from Partition to the bicriteria Spanning Tree, Shortest Path and Perfect Matching problems as presented above, except that we scale the cost of each edge (but not its weight) by a factor of γ. Thus for any solution S, we have that $c(S)/\gamma + w(S) = A$. Let $y = w(S)$, then $c(S) = \gamma(A-y)$. Define $g(y) = f(\gamma(A-y), y) = \gamma^a(A-y)^a + y^b$. Our goal is to choose γ such that the function $g(y)$ is minimized when $y = A/2$. Thus, we want to show that $g'(A/2) = 0$ and $g''(A/2) > 0$. We take the derivative of $g(y)$ and obtain, $g'(y) = -a \cdot \gamma^a(A-y)^{a-1} + by^{b-1}$. Basic calculus shows

$$g'\left(\frac{A}{2}\right) = 0 \iff \gamma = \left(\frac{b}{a}\left(\frac{A}{2}\right)^{b-a}\right)^{\frac{1}{a}} .$$

Finally, we evaluate the second derivative of $g(y)$ at $A/2$ and show that it is positive. We have, $g''(y) = a(a-1)\gamma^a(A-y)^{a-2} + b(b-1)y^{b-2}$. Thus, $g''(A/2) > 0$ when $a > 1$ or $b > 1$.

Observe that, in general, γ is irrational but rounding γ after a polynomial number of bits preserves the desired property. In order to see this, first of all observe that for every $y \in \{0, \ldots, A\}$, $g(y)$ is a rational number whose representation length l is polynomially bounded in the representation length

$\log A$ of A, say $l < p(\log A)$ for some polynomial p. Furthermore, $g(A/2) < g(y)$ for every $y \in \{0,\dots,A\}$ with $y \neq A/2$. Together this implies for every such y, $g(y) - g(A/2) > 2^{-p(\log A)}$. Now let γ_* denote γ rounded after $B = \lceil p(\log A) + 2a(\log A + \log\gamma) + 1\rceil$ bits. We denote by g_* the function $g_*(y) = \gamma_*^a(A-y)^a + y^b$. Then for every $y \in \{0,\dots,A\}$,

$$|g(y) - g_*(y)| = |\gamma^a - \gamma_*^a|(A-y)^a \leq A^a|\gamma^a - \gamma_*^a| \ .$$

Combining this with $|\gamma - \gamma_*| \leq 2^{-B}$ yields

$$|g(y) - g_*(y)| \leq A^a \cdot a(\gamma+1)^a \cdot 2^{-B} < 2^{-p(\log A)-1} \ .$$

Altogether this implies for every $y \in \{0,\dots,A\}$ with $y \neq A/2$

$$g_*(A/2) < g(A/2) + 2^{-p(\log A)-1} < g(y) - 2^{-p(\log A)-1} < g_*(y) \ .$$

This shows that we can replace γ by γ_* without affecting the property that $A/2$ is the unique minimum. $\square$

We will now have a closer look at exponential functions $f(x,y) = 2^{x^\delta} + 2^{y^\delta}$, for some $\delta > 0$. In the following, we assume that there is an oracle, which given two solutions S_1 and S_2, decides in constant time whether $f(c(S_1), w(S_1))$ is larger than $f(c(S_2), w(S_2))$ or vice versa. We show that even in this model of computation there is no polynomial time approximation algorithm with polynomial approximation ratio, unless $P = NP$.

Theorem 2 *Let $f(x,y) = 2^{x^\delta} + 2^{y^\delta}$ with $\delta > 0$. There is no approximation algorithm for the f-*Spanning Tree*, f-*Shortest Path*, and f-*Perfect Matching* problem with polynomial running time and approximation ratio less than $2^{|I|}$, where $|I|$ denotes the input size, unless $P = NP$.*

PROOF. We use the reductions from Partition to the problems we consider as presented in the proof of Theorem 1. Assume that we are given an instance $a_1,\dots,a_n$ of Partition and let $A = \sum_{i=1}^n a_i$. Assume that we scale the natural numbers a_i by a factor of $b > 0$ before constructing the graphs. If there is a desired partition in the original instance, then there is also a solution in the scaled instance with $f(S) = 2^{(b\cdot A/2)^\delta+1}$. If there is no desired partition, then $f(S) \geq 2^{(b\cdot A/2+b)^\delta}$ for any solution S. Obviously, this is a $(2^{(b\cdot A/2)^\delta+1}, 2^{(b\cdot A/2+b)^\delta})$ gap problem for which no polynomial time approximation algorithm with approximation ratio less than $2^{(b\cdot A/2+b)^\delta}/2^{(b\cdot A/2)^\delta+1} = 2^{(b\cdot A/2+b)^\delta-(b\cdot A/2)^\delta-1}$ exists, unless $P = NP$. For arbitrary $C > 0$, choosing

$$b = \left\lceil \left(\frac{C+1}{(A/2+1)^\delta - (A/2)^\delta}\right)^{1/\delta} \right\rceil$$

yields $2^{(b \cdot A/2+b)^\delta - (b \cdot A/2)^\delta - 1} \geq 2^C$.

Due to the scaling, the input size $|I|$ of the constructed graph is increased. Since the constructed graph contains $O(n)$ vertices and edges, we can estimate $|I|$ as follows:

$$\begin{aligned}
|I| &\leq n \cdot (\log A + \log b + \kappa) \\
&\leq n \cdot \left(\log A + \frac{1}{\delta} \left(\log(C+1) - \log((A/2+1)^\delta - (A/2)^\delta) \right) + 1 + \kappa \right)
\end{aligned}$$

for some constant κ.

We first consider the case $\delta \geq 1$. In this case, we have

$$(A/2+1)^\delta - (A/2)^\delta \geq 1$$

and, hence,

$$|I| \leq n \cdot \left(\log A + \frac{1}{\delta} (\log(C) + 2) + 1 + \kappa \right) \quad .$$

For $C \geq A^\delta$, we can estimate this by

$$|I| \leq kn \cdot \log C$$

for an appropriately chosen constant k. Now assume that we had an algorithm $\mathcal{Z}$ with approximation ratio less than $2^{|I|}$. Then

$$2^{|I|} \leq 2^{kn \cdot \log C} = C^{kn} \quad .$$

Hence, setting $C = (kn)^2 A^\delta$ yields that, for sufficiently large n and k, the approximation ratio of $\mathcal{Z}$ is smaller than 2^C. Thus, we can use algorithm $\mathcal{Z}$ to solve the given Partition instance exactly. Note that the length of the input I is polynomially bounded in the size of the Partition instance $a_1, \ldots, a_n$.

Now we consider the case $\delta \leq 1$. In this case, we can use the estimate

$$-\log\left((A/2+1)^\delta - (A/2)^\delta \right) \leq \frac{\log(A/2)}{\delta}$$

to obtain

$$|I| \leq n \cdot \left(\log A + \frac{1}{\delta} \left(\log C + 1 + \frac{\log(A/2)}{\delta} \right) + 1 + \kappa \right) \quad .$$

For $C > A^{1/\delta}$ we can simplify this to

$$|I| \leq \frac{kn}{\delta} \log C$$

for an appropriately chosen constant k. Now assume that we had an algorithm $\mathcal{Z}$ with approximation ratio less than $2^{|I|}$. Then

$$2^{|I|} \leq 2^{\frac{kn}{\delta} \cdot \log C} = C^{\frac{kn}{\delta}} \quad .$$

Hence, setting $C = \frac{(kn)^2 A^{1/\delta}}{\delta}$ yields that, for sufficiently large n and k, the approximation ratio of $\mathcal{Z}$ is smaller than 2^C. Thus, we can use algorithm $\mathcal{Z}$ to solve the given PARTITION instance exactly. Note that the length of the input I is polynomially bounded in the size of the PARTITION instance $a_1, \dots, a_n$. $\square$

2.2 An FPTAS *for a Large Class of Functions*

In this section, we present a sufficient condition for the objective function f under which there is an FPTAS for the f-SPANNING TREE, the f-SHORTEST PATH, and the f-PERFECT MATCHING problem. In fact, our result is not restricted to these problems but applies to every bicriteria optimization problem Π with an FPTAS for approximating the Pareto curve.

We begin by introducing a restricted class of functions f.

Definition 3 *We call a non-decreasing function $f : \mathbb{R}_+^2 \to \mathbb{R}_+$* quasi-polynomially bounded *if there exist constants $c > 0$ and $d > 0$ such that for every $x, y \in \mathbb{R}_+$*

$$\frac{\partial f(x,y)}{\partial x} \cdot \frac{1}{f(x,y)} \leq \frac{c \cdot \ln^d x \cdot \ln^d y}{x}$$

and

$$\frac{\partial f(x,y)}{\partial y} \cdot \frac{1}{f(x,y)} \leq \frac{c \cdot \ln^d x \cdot \ln^d y}{y} \quad .$$

Observe that every non-decreasing polynomial is quasi-polynomially bounded. Furthermore the sum of so-called quasi-polynomial functions of the form $f(x,y) = x^{\text{polylog}(x)} + y^{\text{polylog}(y)}$ is also quasi-polynomially bounded, whereas the sum of exponential functions $f(x,y) = 2^{x^\delta} + 2^{y^\delta}$ is not quasi-polynomially bounded. We are now ready to state our main theorem for this section.

Theorem 4 *There exists an FPTAS for any f-Π problem in which f is non-decreasing and quasi-polynomially bounded if there exists an* FPTAS *for approximating the Pareto curve of* Π.

PROOF. Our goal is to find a solution for the f-Π problem in question with value no more than $(1 + \varepsilon)$ times optimal. The FPTAS for the f-Π problem of relevance is quite simple. It uses the FPTAS for approximating the Pareto curve to generate an ε'-approximate Pareto curve $\mathcal{P}_{\varepsilon'}$ and tests which solution in $\mathcal{P}_{\varepsilon'}$ has the lowest f-value. Recall that the number of points in $\mathcal{P}_{\varepsilon'}$ is polynomial in the size of the input and $1/\varepsilon'$ [13]. The only question to be settled is how small ε' has to be chosen to obtain a $(1+\varepsilon)$-approximation for f-Π by this approach. In particular, we have to show that $1/\varepsilon'$ is polynomially bounded in $1/\varepsilon$ and the input size since then, an ε'-approximate Pareto curve contains only polynomially many solutions and, thus, our approach runs in polynomial time.

Let S^* denote an optimal solution to the f-Π problem. Since f is non-decreasing we can w. l. o. g. assume S^* to be Pareto optimal. We denote by C^* the cost and by W^* the weight of S^*. We know that an ε'-approximate Pareto curve contains a solution S' with cost C' and weight W' such that $C' \leq (1+\varepsilon')C^*$ and $W' \leq (1+\varepsilon')W^*$. We have to choose $\varepsilon' > 0$ such that $f(C', W') \leq (1+\varepsilon)f(C^*, W^*)$ holds, in fact, we will choose ε' such that

$$f((1+\varepsilon') \cdot C^*, (1+\varepsilon') \cdot W^*) \leq (1+\varepsilon) \cdot f(C^*, W^*) \ . \tag{1}$$

In the following, we show that choosing

$$\varepsilon' = \frac{\varepsilon^2}{c2^{d+4} \cdot \ln^{d+1} C \cdot \ln^{d+1} W} \ ,$$

where C denotes sum of all costs $c(e)$ and W denotes the sum of all weights $w(e)$, satisfies inequality (1). Observe that $1/\varepsilon'$ is polynomially bounded in $1/\varepsilon$ and $\ln C^*$ and $\ln W^*$, i. e., the input size.

We start by rewriting $f((1+\varepsilon')C^*, (1+\varepsilon')W^*)$ as follows

$$\begin{aligned} & f((1+\varepsilon') \cdot C^*, (1+\varepsilon') \cdot W^*) \\ = & \begin{cases} f(C^*, W^*)+ \\ f((1+\varepsilon') \cdot C^*, W^*) - f(C^*, W^*)+ \\ f((1+\varepsilon') \cdot C^*, (1+\varepsilon') \cdot W^*) - f((1+\varepsilon') \cdot C^*, W^*) \ . \end{cases} \end{aligned}$$

Now, it is enough to find ε' such that

$$f((1+\varepsilon') \cdot C^*, W^*) - f(C^*, W^*) \leq \frac{\varepsilon}{2} \cdot f(C^*, W^*) \tag{2}$$

and

$$f((1+\varepsilon') \cdot C^*, (1+\varepsilon')W^*) - f((1+\varepsilon') \cdot C^*, W^*) \leq \frac{\varepsilon}{2} \cdot f(C^*, W^*) \ . \tag{3}$$

Before we estimate the terms in (2) and (3) we remind the reader of a version of Bernoulli's inequality which we will use later.

Lemma 5 *Let $x > -1$, $x \in \mathbb{R}$ and $n \in \mathbb{N}$. Then*

$$1 + \frac{x}{n(1+x)} \leq \sqrt[n]{1+x} \leq 1 + \frac{x}{n} \ .$$

Estimating $\mathbf{f((1+\varepsilon')C^*, W^*) - f(C^*, W^*)}$:
We start by estimating the term $f((1+\varepsilon')C^*, W^*) - f(C^*, W^*)$. Therefore we define a function $g : \mathbb{R}_+ \to \mathbb{R}_+$ by $g(x) = f(x, W^*)$. Then we can express the difference we are interested in as $g((1+\varepsilon')C^*) - g(C^*)$. Furthermore, for all $x \in \mathbb{R}_+$, we know

$$\frac{g'(x)}{g(x)} \leq \frac{c \cdot \ln^d x \cdot \ln^d W^*}{x} \ . \tag{4}$$

Let z^* denote $g(C^*) = f(C^*, W^*)$. The difference $g((1+\varepsilon')C^*) - g(C^*)$ becomes maximal when the derivative of g is as large as possible. Thus, we assume w. l. o. g. that inequality (4) is satisfied with equality, i. e.,

$$\frac{g'(x)}{g(x)} = \frac{c \cdot \ln^d x \cdot \ln^d W^*}{x} \ .$$

This differential equation with the additional condition $g(C^*) = z^*$ has a unique solution, namely

$$g(x) = \frac{z^*}{e^{\frac{c}{d+1} \cdot \ln^{d+1} C^* \cdot \ln^d W^*}} e^{\frac{c}{d+1} \cdot \ln^{d+1} x \cdot \ln^d W^*} \ .$$

We want to show $g((1+\varepsilon')C^*) - g(C^*) \leq \varepsilon/2 \cdot g(C^*)$ which is equivalent to $g((1+\varepsilon')C^*)/g(C^*) \leq 1 + \varepsilon/2$. For the sake of simplicity, we assume w. l. o. g. $\varepsilon' < 1$, $C^* \geq e$ and $W^* \geq e$ which implies $\ln(1+\varepsilon') < 1$, $\ln C^* > 1$ and $\ln W^* > 1$. Then we have the following

$$\begin{aligned}
\frac{g((1+\varepsilon')C^*)}{g(C^*)} &= \exp\left(\frac{c}{d+1} \cdot \ln^d W^* (\ln^{d+1}((1+\varepsilon')C^*) - \ln^{d+1} C^*)\right) \\
&= \exp\left(\frac{c}{d+1} \cdot \ln^d W^* \cdot \sum_{i=1}^{d+1} \binom{d+1}{i} \ln^i(1+\varepsilon') \ln^{d+1-i} C^*\right) \\
&\leq \exp\left(\frac{c}{d+1} \cdot \ln^d W^* \cdot 2^{d+1} \ln(1+\varepsilon') \ln^{d+1} C^*\right) \\
&\leq (1+\varepsilon')^{\lceil c 2^{d+1} \cdot \ln^{d+1} C^* \cdot \ln^d W^* \rceil} \ .
\end{aligned}$$

It suffices to show

$$\varepsilon' \leq \left(1 + \frac{\varepsilon}{2}\right)^{\frac{1}{\lceil c 2^{d+1} \cdot \ln^{d+1} C^* \cdot \ln^d W^* \rceil}} - 1$$

since this implies

$$(1+\varepsilon')^{\lceil c2^{d+1}\cdot \ln^{d+1} C^* \cdot \ln^d W^* \rceil} \le 1+\frac{\varepsilon}{2} \enspace .$$

We can apply Lemma 5 to obtain

$$\begin{aligned}
&\varepsilon' \le \frac{\varepsilon/2}{\lceil c2^{d+1}\cdot \ln^{d+1} C^* \cdot \ln^d W^* \rceil (1+\varepsilon/2)}\\
\Rightarrow\, &\varepsilon' \le \left(1+\frac{\varepsilon}{2}\right)^{\frac{1}{\lceil c2^{d+1}\cdot \ln^d W^* \cdot \ln^{d+1} C^* \rceil}} - 1.
\end{aligned}$$

Thus, choosing

$$\varepsilon' = \frac{\varepsilon}{c2^{d+4}\cdot \ln^{d+1} C^* \cdot \ln^d W^*} \tag{5}$$

yields $g((1+\varepsilon')C^*) - g(C^*) \le \varepsilon/2 \cdot g(C^*)$.

Estimating $\mathbf{f}((\mathbf{1}+\varepsilon')\mathbf{C}^*, (\mathbf{1}+\varepsilon')\mathbf{W}^*) - \mathbf{f}((\mathbf{1}+\varepsilon')\mathbf{C}^*, \mathbf{W}^*)$:
Now define $h : \mathbb{R}_+ \to \mathbb{R}_+$ by $h(y) = f((1+\varepsilon')C^*, y)$. Observe that we can use the arguments we used in the previous paragraph to show $h((1+\varepsilon')W^*) - h(W^*) \le \varepsilon/2 \cdot h(W^*)$ for an analogously chosen ε' but this is not enough since $h(W^*) = f((1+\varepsilon')C^*, W^*) \ge f(C^*, W^*)$.

Following the arguments of the last paragraph we can show that setting

$$\varepsilon' = \frac{\varepsilon^2}{c2^{d+4}\cdot \ln^{d} C^* \cdot \ln^{d+1} W^*} \tag{6}$$

yields

$$f((1+\varepsilon')C^*, (1+\varepsilon')W^*) - f((1+\varepsilon')C^*, W^*) \le \frac{\varepsilon^2}{2} f((1+\varepsilon')C^*, W^*) \enspace . \tag{7}$$

We assume w.l.o.g. $\varepsilon < 0.7$. Then, a second application of the result of the last paragraph shows

$$\begin{aligned}
& f((1+\varepsilon')C^*, W^*) - f(C^*, W^*) &&\le \frac{\varepsilon}{2} f(C^*, W^*)\\
\Rightarrow \quad & f((1+\varepsilon')C^*, W^*) &&\le \frac{2+\varepsilon}{2} f(C^*, W^*)\\
\Rightarrow \quad & \frac{2}{2+\varepsilon} f((1+\varepsilon')C^*, W^*) &&\le f(C^*, W^*)\\
\Rightarrow \quad & \varepsilon f((1+\varepsilon')C^*, W^*) &&\le f(C^*, W^*) \enspace ,
\end{aligned} \tag{8}$$

where the last inequality follows from the assumption $\varepsilon < 0.7$. Putting together inequalities (7) and (8) yields

$$f((1+\varepsilon')C^*, (1+\varepsilon')W^*) - f((1+\varepsilon')C^*, W^*) \leq \frac{\varepsilon^2}{2} f((1+\varepsilon')C^*, W^*)$$
$$\leq \frac{\varepsilon}{2} f(C^*, W^*) \ .$$

Observe that the choice of ε' in (5) and (6) is dependent on the cost C^* and the weight W^* of an optimal solution. These values are unknown but can be upper bounded by C and W the sum of all costs $c(e)$ respectively all weights $w(e)$. Thus, in (5) and (6) we can replace C^* by C and W^* by W and choose

$$\varepsilon' = \frac{\varepsilon^2}{c2^{d+4} \cdot \ln^{d+1} C \cdot \ln^{d+1} W} \ . \qquad \square$$

Observe that Theorem 4 is almost tight since for every $\delta > 0$ we can construct a function f for which the quotients of the partial derivatives and $f(x, y)$ are lower bounded by $\delta/x^{1-\delta}$ respectively by $\delta/y^{1-\delta}$ and for which the f-Π problem does not posses an FPTAS, namely $f(x, y) = 2^{x^\delta} + 2^{y^\delta}$.

3 Smoothed Analysis of Bicriteria Problems

In the previous section, we have shown that f-Π problems are NP-hard even for simple polynomial objective functions and we have also shown that it is even hard to approximate them for rapidly increasing objective functions if Π is either the bicriteria Spanning Tree, Shortest Path or Perfect Matching problem. In this section, we will analyze f-Π problems in a probabilistic input model rather than from a worst-case viewpoint. In this model, we show that, for every $p > 0$ for which $1/p$ is polynomial in the input size, the f-Π problem can be solved in polynomial time for *every* non-decreasing objective function with probability $1 - p$ if there exists a pseudopolynomial time algorithm for generating the Pareto set of Π. It is known that for the bicriteria graph problems we deal with the expected size of the Pareto set in the considered probabilistic input model is polynomially bounded [3]. Thus, if we had an algorithm for generating the set of Pareto optimal solutions whose running time is bounded polynomially in the input size and the number of Pareto optimal solutions, then we could, for any non-decreasing objective function f, devise an algorithm for the f-Π problem that is efficient on semi-random inputs.

For a few problems, e. g., the Shortest Path problem, efficient (w. r. t. the input size and the size of the Pareto set) algorithms for generating the Pareto set are known [18,7]. But it is still unknown whether such an algorithm exists for the Spanning Tree or the Perfect Matching problem, whereas it is known that there exist for, e. g., the SPANNING TREE and the PERFECT MATCHING problem, pseudopolynomial time algorithms (w. r. t. costs and weights) for generating the reduced Pareto set. This follows since the exact versions of the single objective versions of these problems, i. e., the question, "Is there a spanning tree/perfect matching with cost exactly c?", can be solved in pseudopolynomial time (w. r. t. the costs) [2,11,12]. We will show how such pseudopolynomial time algorithms can be turned into algorithms for efficiently generating the Pareto set of semi-random inputs.

3.1 Probabilistic Input Model

Usually, the input model considered in smoothed analysis consists of two stages: First an adversary chooses an input instance, then this input is randomly perturbed in the second stage. For the bicriteria graph problems considered in this paper, the input given by the adversary is a graph $G = (V, E, w, c)$ with weights $w : E \to \mathbb{R}_+$ and costs $c : E \to \mathbb{R}_+$, and in the second stage these weights and costs are perturbed by adding independent random variables to them.

We can replace this two-step model by a one-step model where the adversary is only allowed to specify a graph $G = (V, E)$ and, for each edge $e \in E$, two probability distributions, namely one for $c(e)$ and one for $w(e)$. The costs and weights are then independently drawn according to the given probability distributions. Of course, the adversary is not allowed to specify arbitrary distributions since this would include deterministic inputs as a special case. We place two restrictions upon the distributions concerning the expected value and the maximal density. To be more precise, for each weight and each cost, the adversary is only allowed to specify a distribution which can be described by a piecewise continuous density function $f : \mathbb{R}_+ \to \mathbb{R}_+$ with expected value at most 1 and maximal density at most ϕ, i. e., $\int_{x\in\mathbb{R}_+} x f(x)\,dx \le 1$ and $\sup_{x\in\mathbb{R}_+} f(x) = \phi$, for a given $\phi \ge 1$.

Observe that restricting the expected value to be at most 1 is without loss of generality, since we are only interested in the Pareto set which is not affected by scaling weights and costs. The parameter ϕ can be seen as a parameter specifying how close the analysis is to a worst case analysis. The larger ϕ the more concentrated the probability distribution can be. Thus, the larger ϕ, the more influence the adversary has. We will call inputs created by this probabilistic input model *ϕ-perturbed inputs.*

Spielman and Teng use Gaussian perturbations in their smoothed analysis of the simplex algorithm to model random noise [14]. Observe that Gaussian distributions with standard deviation σ are a special case of our input model with $\phi = 1/(\sigma\sqrt{2\pi})$.

Note that the costs and weights are irrational with probability 1 since they are chosen according to continuous probability distributions. We ignore their contribution to the input length and assume that the bits of these coefficients can be accessed by asking an oracle in time $O(1)$ per bit. Thus, in our case only the representation of the graph $G = (V, E)$ determines the input length. In the following, let m denote the number of edges, i. e., $m = |E|$.

We assume that there do not exist two different solutions S and S' with either $w(S) = w(S')$ or $c(S) = c(S')$. We can assume this without loss of generality since in our probabilistic input model, two such solutions exist only with probability 0.

3.2 Generating the Pareto set

In this section, we will show how a pseudopolynomial time algorithm $\mathcal{A}$ for generating the Pareto set can be turned into a polynomial time algorithm which succeeds with probability at least $1-p$ on semi-random inputs, for any given $p > 0$ where $1/p$ is polynomial in the input size. In order to apply $\mathcal{A}$ efficiently it is necessary to round the costs and weights, such that they are only polynomially large after the rounding, i. e., such that the lengths of their representations are only logarithmic in the input size. Let $\lfloor c \rfloor_b$ and $\lfloor w \rfloor_b$ denote the costs and weights rounded down to the b-th bit after the binary point. We denote by $\mathcal{P}$ the Pareto set of the ϕ-perturbed input $G = (V, E, w, c)$ and by $\mathcal{P}_b$ the Pareto set of the rounded ϕ-perturbed input $G = (V, E, \lfloor w \rfloor_b, \lfloor c \rfloor_b)$.

Theorem 6 *For $b = \Theta\left(\log\left(\frac{m\phi}{p}\right)\right)$, $\mathcal{P}$ is a subset of $\mathcal{P}_b$ with probability at least $1-p$.*

This means, we can round the coefficients after only a logarithmic number of bits and use the pseudopolynomial time algorithm, which runs on the rounded input in polynomial time, to obtain $\mathcal{P}_b$. With probability at least $1-p$ the set $\mathcal{P}_b$ contains all Pareto optimal solutions from $\mathcal{P}$ but it can contain solutions which are not Pareto optimal w. r. t. w and c. By removing these superfluous solutions we obtain the set $\mathcal{P}$ with probability at least $1-p$.

Corollary 7 *There exists an algorithm for generating the Pareto set of* Π *on ϕ-perturbed inputs with failure probability at most p and running time* $\mathrm{poly}(m, \phi, 1/p)$*, if there exists a pseudopolynomial time algorithm for generating the reduced Pareto set of* Π.

From the definition of a Pareto optimal solution, it follows that the optimal solution S of a constrained problem, i. e., the weight-minimal solution among

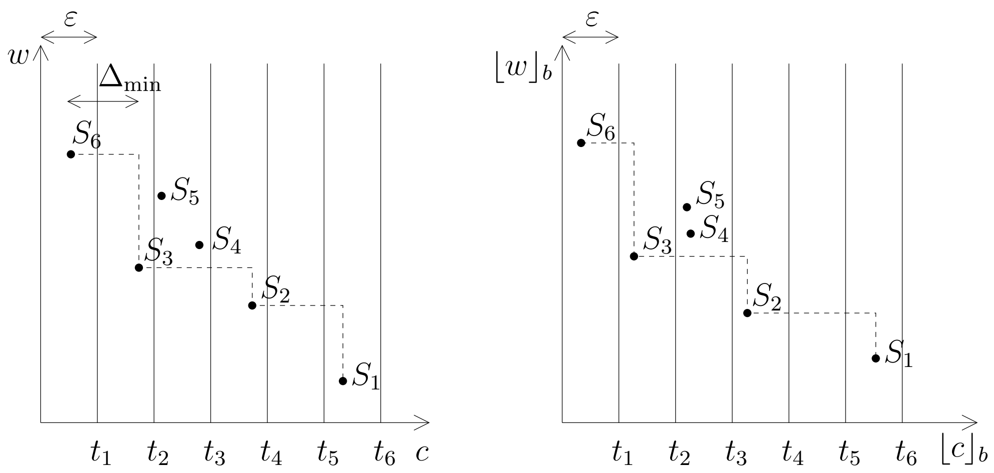

Fig. 3. Successful case: $\varepsilon < \Delta_{\min}$, $S^{(1)} = S_b^{(1)} = S_6$, $S^{(2)} = S_b^{(2)} = S^{(3)} = S_b^{(3)} = S_3$, $S^{(4)} = S_b^{(4)} = S^{(5)} = S_b^{(5)} = S_2$, $S^{(6)} = S_b^{(6)} = S_1$, $c(S_i) \leq z \cdot \varepsilon = 6 \cdot \varepsilon$.

all solutions fulfilling a cost constraint $c(S) \leq t$, is always a Pareto optimal solution. This is because, if there were a solution S' that dominates S, then S' would also be a better solution to the constrained problem. We will show that, for every $S \in \mathcal{P}$, with sufficiently large probability we can find a threshold t such that S is the optimal solution to the constrained problem $\min \lfloor w \rfloor_b(S)$ w. r. t. $\lfloor c \rfloor_b(S) \leq t$, i. e., with sufficiently large probability every $S \in \mathcal{P}$ is Pareto optimal w. r. t. the rounded coefficients.

To be more precise, we consider, for appropriately chosen z and ε, z many constrained problems with weights $\lfloor w \rfloor_b$, costs $\lfloor c \rfloor_b$ and thresholds $t_i = i \cdot \varepsilon$, for $i \in [z] := \{1, 2, \ldots, z\}$. We will denote the minimal cost difference between two different Pareto optimal solutions by $\Delta_{\min}$, i. e.,

$$\Delta_{\min} = \min_{\substack{S_1, S_2 \in \mathcal{P} \\ S_1 \neq S_2}} |c(S_1) - c(S_2)| \ .$$

If $\Delta_{\min}$ is larger than ε, then $\mathcal{P}$ consists only of solutions to constrained problems of the form $\min w(S)$, w. r. t. $c(S) \leq t_i$, since, if $\varepsilon < \Delta_{\min}$, we do not miss a Pareto optimal solution by our choice of thresholds. Based on results by Beier and Vöcking [4], we will prove that, for each $i \in [z]$, the solution $S^{(i)}$ to the constrained problem $\min w(S)$ w. r. t. $c(S) \leq t_i$ is the same as the solution $S_b^{(i)}$ to the constrained problem $\min \lfloor w \rfloor_b(S)$ w. r. t. $\lfloor c \rfloor_b(S) \leq t_i$ with sufficiently large probability. Thus, if $\varepsilon < \Delta_{\min}$ and $S^{(i)} = S_b^{(i)}$ for all $i \in [z]$, then $\mathcal{P} \subseteq \mathcal{P}_b$. See Figure 3 for an illustration of this approach.

We do not know how to determine $\Delta_{\min}$ in polynomial time, but we can show a lower bound ε for $\Delta_{\min}$ that holds with a certain probability. Based on this lower bound, we can appropriately choose ε. We must choose z sufficiently large, so that $c(S) \leq z \cdot \varepsilon$ holds with sufficiently high probability for every solution S. Thus, our analysis fails only if one of the following three failure

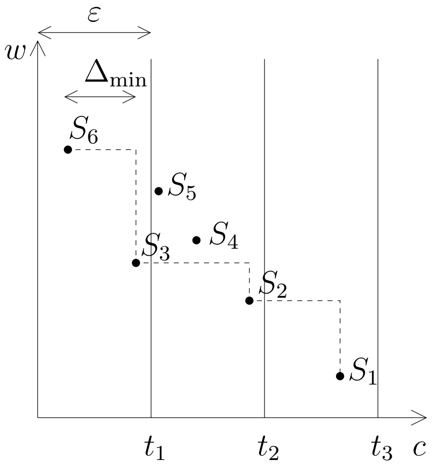

Fig. 4. Failure event $\mathcal{F}_1$: $\varepsilon > \Delta_{\min}$. S_6 is not a solution to any of the constrained problems.

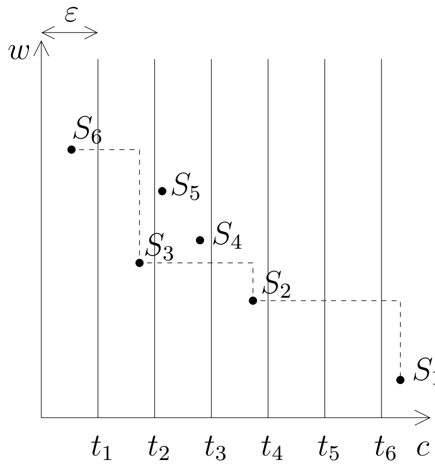

Fig. 5. Failure event $\mathcal{F}_3$: $c(S_1) > z \cdot \varepsilon = 6 \cdot \varepsilon$. S_1 is not a solution to any of the constrained problems.

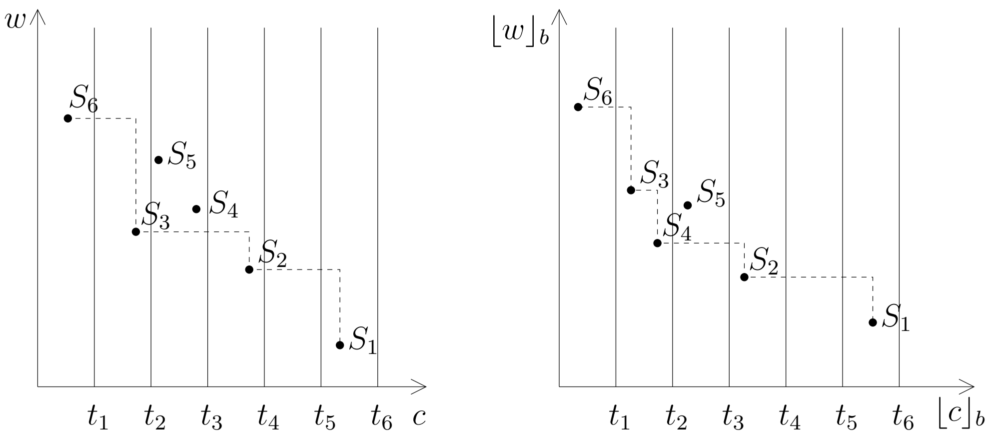

Fig. 6. Failure event $\mathcal{F}_2$: $S^{(2)} = S_3 \neq S_4 = S_b^{(2)}$.

events occurs (see also Figures 4, 5, and 6):

- $\mathcal{F}_1$: $\Delta_{\min}$ is smaller than the chosen ε.
- $\mathcal{F}_2$: For one $i \in [z]$, the solution $S^{(i)}$ to $\min w(S)$ w. r. t. $c(S) \leq t_i$ does not equal the solution $S_b^{(i)}$ to $\min \lfloor w \rfloor_b(S)$ w. r. t. $\lfloor c \rfloor_b(S) \leq t_i$.
- $\mathcal{F}_3$: There exists a solution S with $c(S) > z \cdot \varepsilon$.

In order to prove Theorem 6, we first have to estimate the probability of the failure events $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$. Depending on these failure probabilities, we can choose appropriate values for z, ε and b yielding the theorem. We start by estimating the probability of the first failure event, which is the most involved part of the proof. The probability of $\mathcal{F}_2$ is estimated directly by using a result of Beier and Vöcking [3], and the probability of $\mathcal{F}_3$ is estimated by a simple application of Markov's inequality.

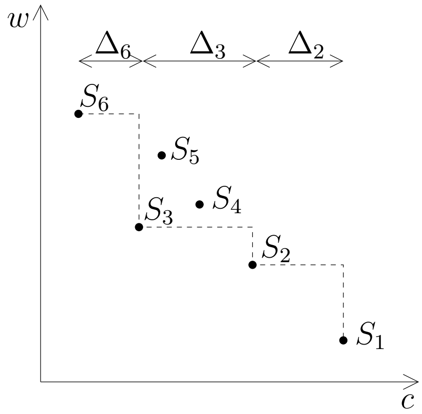

Fig. 7. $\Delta_4 = \Delta_5 = 0$ and $\Delta_{\min} = \Delta_6$.

3.3 Estimating the Size of the Smallest Gap

In order to bound the probability of the failure event $\mathcal{F}_1$, we first write Π as a binary program. We introduce a variable $x_e \in \{0,1\}$ for every edge $e \in E$, and we denote by $\mathcal{S} \subseteq \{0,1\}^m$ the set of all solutions of Π, e. g., the set of all spanning trees or all perfect matchings of G. For bounding $\Delta_{\min}$, it is not necessary that the weights are chosen at random, since the bound we will prove holds for every deterministic choice of the weights. Thus, we assume the weights to be fixed arbitrarily.

Now let $S_1, \ldots, S_l$ denote a sequence containing all elements from $\mathcal{S}$ ordered such that $w(S_1) \leq \ldots \leq w(S_l)$. For $j \in \{2, \ldots, l\}$, we define $\Delta_j = \min_{i \in [j-1]} c(S_i) - \min_{i \in [j]} c(S_i)$. Observe that a solution S_j, for $j \in \{2, \ldots, l\}$, is Pareto optimal if and only if $\Delta_j > 0$ and that Δ_j describes how much less S_j costs compared to the cheapest solution S_i with $i < j$ (see Figure 7). Thus, we can write $\Delta_{\min}$ as follows

$$\Delta_{\min} = \min_{j \in [l] \setminus \{1\}} \{\Delta_j | \Delta_j > 0\} \ .$$

Our goal is to bound the probability that $\Delta_{\min}$ lies below a given value ε. Therefore, we rewrite $\mathbf{Pr}[\Delta_{\min} < \varepsilon]$ as follows:

$$\begin{aligned} \mathbf{Pr}[\Delta_{\min} < \varepsilon] &= \mathbf{Pr}[\exists j \in [l] \setminus \{1\} : 0 < \Delta_j < \varepsilon] \\ &\leq \sum_{j \in [l] \setminus \{1\}} \mathbf{Pr}[\Delta_j > 0] \cdot \mathbf{Pr}[\Delta_j < \varepsilon | \Delta_j > 0] \ . \end{aligned} \tag{9}$$

Assume, we could bound $\mathbf{Pr}[\Delta_j < \varepsilon | \Delta_j > 0]$ from above for every j by some term a. Then we would have

$$
\begin{aligned}
\mathbf{Pr}\left[\Delta_{\min} < \varepsilon\right] &\leq a \cdot \sum_{j\in[l]\setminus\{1\}} \mathbf{Pr}\left[\Delta_j > 0\right] \\
&= a \cdot (\mathbf{E}\left[q\right] - 1) \\
&\leq a \cdot \mathbf{E}\left[q\right] \quad ,
\end{aligned}
$$

where q denotes the number of Pareto optimal solutions.

In this scenario, we can apply the analysis of Beier and Vöcking to obtain a polynomial upper bound on the expected number of Pareto optimal solutions [3]. The crucial point in their analysis is a lower bound on $\mathbf{E}\left[\Delta_j | \Delta_j > 0\right]$ for every $j \in [l] \setminus \{1\}$. Unfortunately, we cannot apply their results directly to bound the conditional probability $\mathbf{Pr}\left[\Delta_j < \varepsilon | \Delta_j > 0\right]$ since, in general, a bound on the conditional expectation does not imply a bound on the conditional probability. Nonetheless, we prove the following result.

Theorem 8 *Let the costs be independent, positive random variables whose expectations are bounded by* 1 *and whose densities are bounded by* ϕ*, i. e., for all* $x \in \mathbb{R}_+$ *and for all* $e \in E$ *it holds* $f_e(x) \leq \phi$*. Then, for* $m = |E|$ *and* $\varepsilon \leq (6m^8\phi^2)^{-1}$,

$$\mathbf{Pr}\left[\Delta_{\min} < \varepsilon\right] \leq 2(6\varepsilon m^5 \phi^2)^{1/3} \quad .$$

Analogously to the analysis in [3], we also look at long-tailed distributions first and, after that, use the results for long-tailed distributions to analyze the general case.

3.3.1 Long-Tailed Distributions

One can classify continuous probability distributions by comparing their tails with the tail of the exponential distribution. In principle, if the tail function of a distribution can be lower bounded by the tail function of an exponential function, then we say the distribution has a "long tail".

Of special interest to us is the behavior of the tail function under a logarithmic scale. Given any continuous probability distribution with density $f : \mathbb{R}_+ \to \mathbb{R}_+$, the *tail function* $T : \mathbb{R}_+ \to [0, 1]$ is defined by $T(t) = \int_t^\infty f(x)\,dx$. We define the *slope* of T at $x \in \mathbb{R}_+$ to be the first derivative of the function $-\ln(T(\cdot))$ at x, i. e., $\mathrm{slope}_T(x) = -[\ln(T(x))]'$. For example, the tail function of the exponential distribution with parameter λ is $T(x) = \exp(-\lambda x)$ so that the slope of this function is $\mathrm{slope}_T(x) = \lambda$, for every $x \geq 0$. The tail of a continuous probability distribution is defined to be *long* if there exists a constant $\alpha > 0$ such that $\mathrm{slope}_T(x) \leq \alpha$, for every $x \geq 0$.

We denote by T_e the tail function of $c(e)$ and by f_e the corresponding density. Beier and Vöcking prove the following theorem on the expected number of Pareto optimal solutions.

Theorem 9 ([3]) *Let $c(e)$ be a positive, long-tailed random variable with expected value at most μ, for each $e \in E$, and let α be a positive real number satisfying* $\mathrm{slope}_{T_e}(x) \leq \alpha$, *for every $x \geq 0$ and every $e \in E$. Finally, let q denote the number of Pareto optimal solutions and let $m = |E|$. Then*

$$\mathbf{E}[q] \leq \alpha\mu m^2 + 1 \leq 2\alpha\mu m^2 \ .$$

In order to bound the conditional probability $\mathbf{Pr}[\Delta_j < \varepsilon | \Delta_j > 0]$, we have to take a closer look at the proof of Theorem 9. The following lemma is implicitly contained in this proof.

Lemma 10 ([3]) *Let α and μ as in Theorem 9, then, for every $j \in [l]$ and for $m = |E|$, it holds*

$$\mathbf{Pr}[\Delta_j < \varepsilon | \Delta_j > 0] \leq 1 - \exp(-m\alpha\varepsilon) \ .$$

Let $\varepsilon \geq 0$ be fixed arbitrarily. Combining Theorem 9 and Lemma 10 with equation (9) yields

$$\begin{aligned}
\mathbf{Pr}[\Delta_{\min} < \varepsilon] &\leq \sum_{j \in [l] \setminus \{1\}} \mathbf{Pr}[\Delta_j > 0] \cdot \mathbf{Pr}[\Delta_j < \varepsilon | \Delta_j > 0] \\
&\leq (1 - \exp(-m\alpha\varepsilon)) \cdot \mathbf{E}[q] \\
&\leq \varepsilon \cdot m\alpha \cdot \mathbf{E}[q] \\
&\leq \varepsilon \cdot 2m^3\alpha^2\mu \ .
\end{aligned}$$

Thus, we obtain the following lemma.

Lemma 11 *For each $e \in E$, let $c(e)$ be a positive, long-tailed random variable with expected value at most μ and let α be a positive real number satisfying* $\mathrm{slope}_{T_e}(x) \leq \alpha$, *for every $x \geq 0$ and every $e \in E$. Then, for every $\varepsilon \geq 0$ and for $m = |E|$, it holds*

$$\mathbf{Pr}[\Delta_{\min} < \varepsilon] \leq \varepsilon \cdot 2m^3\alpha^2\mu \ .$$

3.3.2 General Distributions with Bounded Mean and Bounded Density

For general distributions, a statement like Lemma 10 is not true anymore. Nonetheless, Beier and Vöcking were able to bound the expected number of Pareto optimal solutions for any continuous distribution with bounded mean and bounded density.

Theorem 12 ([3]) *Let the costs be independent, positive random variables whose expectations are bounded by μ and whose densities are bounded by ϕ,* *i. e., for all $x \in \mathbb{R}_+$ and for all $e \in E$, it holds $f_e(x) \leq \phi$. Then for $m = |E|$,*

$$\mathbf{E}[q] = O(\phi\mu m^4) \ .$$

We will use Theorem 12 to prove the following bound for $\Delta_{\min}$ which contains Theorem 8 as a special case.

Theorem 13 *Let μ, ϕ, and m as in Theorem 12. Then for $\varepsilon \leq (6m^8\phi^2\mu)^{-1}$,*

$$\mathbf{Pr}[\Delta_{\min} < \varepsilon] \leq 2(6\varepsilon m^5\phi^2\mu)^{1/3} \ .$$

PROOF. For every edge $e \in E$, we define a random variable $x_e = T_e(c(e))$. For any $a > 0$, let $\mathcal{F}_a$ denote the event that, for at least one edge $e \in E$, it holds $x_e \leq a$. We will show, that we can apply the analysis for long-tailed distributions, if $\mathcal{F}_a$ does not occur. We obtain

$$\mathbf{Pr}[\Delta_{\min} < \varepsilon] \leq \mathbf{Pr}[\mathcal{F}_a] + \mathbf{Pr}[\Delta_{\min} < \varepsilon \wedge \neg\mathcal{F}_a] \ . \tag{10}$$

Observe that the x_e's are uniformly distributed over $[0, 1]$, as

$$\begin{aligned}
\mathbf{Pr}[x_e \leq z] &= \mathbf{Pr}\left[c(e) \geq T_e^{-1}(z)\right] \\
&= \int_{T_e^{-1}(z)}^{\infty} f_e(x)\, dx \\
&= T_e(T_e^{-1}(z)) = z \ .
\end{aligned}$$

Thus, we obtain

$$\mathbf{Pr}[\mathcal{F}_a] = \mathbf{Pr}[\exists e \in E : x_e \leq a] \leq ma \ . \tag{11}$$

We would like to estimate $\mathbf{Pr}[\Delta_{\min} < \varepsilon \wedge \neg\mathcal{F}_a]$ in such a way that we get rid of the event $\neg\mathcal{F}_a$, since, under the condition $\neg\mathcal{F}_a$, the random variables $c(e)$ are short-tailed instead of long-tailed. If the event $\mathcal{F}_a$ does not occur, the distribution of $c(e)$ for values larger than $T_e^{-1}(a)$ is not important, thus, we can replace the tail function T_e by a tail function T_e^* with

$$T_e^*(x) = \begin{cases} T_e(x) & \text{if} \qquad x \leq T_e^{-1}(a) \\ a \cdot \exp(-\phi m(x - T_e^{-1}(a)) & \text{otherwise} \end{cases} \ .$$

We denote by $c^*(e)$ a random variable drawn according to the tail function T_e^*. Furthermore, we denote by $\Delta^*_{\min}$ the random variable equivalent to $\Delta_{\min}$ but w. r. t. the costs $c^*(e)$ drawn according to the tail functions T_e^* instead of T_e, and obtain

$$\mathbf{Pr}[\Delta_{\min} < \varepsilon \wedge \neg\mathcal{F}_a] = \mathbf{Pr}[\Delta^*_{\min} < \varepsilon \wedge \neg\mathcal{F}_a] \leq \mathbf{Pr}[\Delta^*_{\min} < \varepsilon] \ . \tag{12}$$

Let f_e^* denote a density corresponding to the tail function T_e^*. The random variable $c^*(e)$ is long-tailed, as the following calculation shows:

$$\mathrm{slope}_{T_e^*}(x) = -\frac{d}{dx}\ln(T_e^*(x)) = \frac{f_e^*(x)}{T_e^*(x)} \le \begin{cases} \phi/a \text{ if} & x \le T_e^{-1}(a) \\ \phi m \text{ otherwise} \end{cases} .$$

For $a \le 1/m$, we obtain

$$\mathrm{slope}_{T_e^*}(x) \le \phi/a \ .$$

Before we can apply Lemma 11, we have to calculate the expectations of the random variables $c^*(e)$ drawn according to the tail functions T_e^*. It holds

$$\begin{aligned} \int_0^\infty x f_e^*(x)\,dx &= \int_0^{T_e^{-1}(a)} x f_e(x)\,dx + \int_{T_e^{-1}(a)}^\infty x f_e^*(x)\,dx \\ &\le \mu + a\phi m \int_0^\infty (x + T_e^{-1}(a))e^{-\phi m x}\,dx \\ &\le \mu + a\phi m \left(\int_0^\infty x e^{-\phi m x}\,dx + \int_0^\infty T_e^{-1}(a) e^{-\phi m x}\,dx\right) \\ &\le \mu + \frac{a}{\phi m} + a \cdot T_e^{-1}(a) \ . \end{aligned}$$

An application of Markov's inequality yields $T_e(a) = \mathbf{Pr}\left[c(e) \ge a\right] \le \mu/a$, and, hence, also $T_e^{-1}(a) \le \mu/a$. Therefore, we have

$$\int_0^\infty x f_e^*(x)\,dx \le \mu + \frac{a}{\phi m} + \mu \le 2\mu + 1 \le 3\mu \ .$$

Applying Lemma 11 with $\alpha' = \phi/a$ and $\mu' = 3\mu$ yields, for every $\varepsilon \ge 0$,

$$\mathbf{Pr}\left[\Delta_{\min}^* < \varepsilon\right] \le \frac{6\varepsilon m^3 \phi^2 \mu}{a^2} \ . \tag{13}$$

Equations (10) to (13) result in the following bound

$$\mathbf{Pr}\left[\Delta_{\min} < \varepsilon\right] \le ma + \frac{6\varepsilon m^3 \phi^2 \mu}{a^2} \ .$$

We choose $a = (6\varepsilon m^2 \phi^2 \mu)^{1/3}$ and obtain

$$\mathbf{Pr}\left[\Delta_{\min} < \varepsilon\right] \le 2(6\varepsilon m^5 \phi^2 \mu)^{1/3} \ .$$

We assumed a to be less or equal to $1/m$, thus, we have to choose ε such that $(6\varepsilon m^5\phi^2\mu)^{1/3} \le 1/m$. This is equivalent to $\varepsilon \le (6m^8\phi^2\mu)^{-1}$. $\square$

In the following, fix some $i \in [z]$ and let $\mathcal{F}_2^{(i)}$ denote the event that the solution $S^{(i)}$ does not equal the solution $S_b^{(i)}$. The situation is very similar to the situation considered in [4]: We have a linear binary optimization problem and we need to bound the probability that rounding the coefficients in the objective function and the constraint changes the optimal solution. In order to bound this probability, Beier and Vöcking introduce and analyze three structural properties, called *winner, loser*, and *feasibility gap*.

Let S^* denote the optimal solution of the constraint problem $\min w(S)$ w. r. t. $c(S) \leq t_i$ and $S \in \mathcal{S}$ and let S^{**} denote the second best solution. The *winner gap* Δ denotes the difference in the objective values of S^* and S^{**}, i. e., $\Delta = w(S^{**}) - w(S^*)$. The *feasibility gap* Γ denotes the slack of S^* w. r. t. the threshold t_i, i. e., $\Gamma = t_i - c(S^*)$. We call a solution $S \in \mathcal{S}$ a *loser* if its objective value is better than that of S^* but it is not feasible due to the linear constraint, i. e., $w(S) \leq w(S^*)$ and $c(S) > t_i$. Let $\mathcal{L}$ denote the set of losers and let the loser gap Λ denote the distance of $\mathcal{L}$ from the threshold t_i, i. e., $\Lambda = \min_{S \in \mathcal{L}} c(S) - t_i$.

The crucial observation in Beier and Vöcking's analysis is that, whenever winner, loser, and feasibility gap are large, the optimal solution of the constraint problem stays optimal even w. r. t. the rounded coefficients. In order to see this, observe that rounding down a coefficient after the b-th bit lowers its value by at most 2^{-b}. Hence the total cost and the total weight of any solution S is decreased due to the rounding by at most $m2^{-b}$. Assume that we first round the costs and consider the intermediate problem with rounded costs but unrounded weights. The optimal solution to this intermediate problem can only deviate from the optimal solution S^* of the original problem if the loser gap Λ is smaller than $m2^{-b}$ since otherwise no solution with smaller weight than S^* becomes feasible due to the rounding. Now consider the problem with rounded weights and costs and an optimal solution to this problem. This solution can only deviate from the optimal solution of the intermediate problem if the winner gap Δ of the intermediate problem is smaller than $m2^{-b}$.

Hence, if S^* is not the optimal solution w. r. t. the rounded coefficients, then either the loser gap Λ is smaller than $m2^{-b}$ or the winner gap Δ of the intermediate problem is smaller than $m2^{-b}$. The following bounds on the probabilities of these events are shown in [4].

Lemma 14 ([4]) *Let $m = |E|$ and let ϕ denote the maximal density of the probability distributions for the costs and weights. For every $\varepsilon \geq 0$,*

$$\mathbf{Pr}\left[\Delta \leq \varepsilon\right] \leq 2\phi m \cdot \varepsilon$$

and

$$\mathbf{Pr}\left[\Lambda \le \varepsilon\right] \le \phi m^2 \cdot \varepsilon \ \ .$$

Using this lemma and the previous observations yields the following theorem.

Theorem 15 *For every $i \in [z]$, $\mathbf{Pr}\left[\mathcal{F}_2^{(i)}\right] \le 2^{-b+2} m^3 \phi$.*

PROOF. Combining our previous observations with Lemma 14 yields

$$\begin{aligned}\mathbf{Pr}\left[\mathcal{F}_2^{(i)}\right] &\le \mathbf{Pr}\left[\Delta \le m2^{-b}\right] + \mathbf{Pr}\left[\Lambda \le m2^{-b}\right] \\ &\le m^2 \cdot 2^{-b+1} \cdot \phi + m^3 \cdot 2^{-b} \cdot \phi \\ &\le 2^{-b+2} m^3 \phi \ \ .\end{aligned}$$

□

By applying a union bound, we get the following corollary.

Corollary 16 $\mathbf{Pr}\left[\mathcal{F}_2\right] \le z \cdot 2^{-b+2} m^3 \phi$.

Now we use Theorem 8 and Corollary 16 to prove Theorem 6.

PROOF of Theorem 6. We would like to choose ε, z and b in such a way that each of the failure probabilities $\mathbf{Pr}\left[\mathcal{F}_i\right]$ is bounded by $p/3$. By Theorem 8, choosing $\varepsilon = p^3(1296 m^5 \phi^2)^{-1}$ yields $\mathbf{Pr}\left[\mathcal{F}_1\right] \le p/3$. By a simple application of Markov's inequality, we obtain that choosing

$$z = \frac{3888 m^6 \phi^2}{p^4}$$

implies $\mathbf{Pr}\left[\mathcal{F}_3\right] \le p/3$. With Corollary 16 we obtain, that setting $b = \log(\alpha m^9 \phi^3 / p^5)$, for an appropriate constant α, yields $\mathbf{Pr}\left[\mathcal{F}_2\right] \le p/3$.

This proves the theorem, since for $b = \log(\alpha m^9 \phi^3/p^5) = \Theta(\log(m\phi/p))$ the failure probability is at most p. □

4 Conclusions

We considered two approaches to bicriteria optimization problems Π, namely generating the Pareto set and solving the decision maker's problem f-Π. In particular, we developed algorithms to the decision maker's problem based on approximate and smoothed Pareto sets. We showed that there is an FPTAS for every f-Π problem if f is quasi-polynomially bounded and if there is a pseudo-polynomial time algorithm for generating the Pareto set. To bypass the limitations of approximate decision making, we turned our attention to decision making in the context of smoothed analysis. We showed how a deterministic algorithm for generating the Pareto set with pseudopolynomial running time can be turned into an algorithm for generating the smoothed Pareto set with small failure probability and polynomial running time. We left open the question whether there is a deterministic algorithm for generating the Pareto set of the SPANNING TREE or PERFECT MATCHING problem whose running time is polynomially bounded in the size of the Pareto set.

Finally, let us remark that all results about the (in)approximability of f-Π problems from Section 2 can be canonically generalized to problems with more than 2 dimensions. However, we do not know whether our results on the smoothed complexity from Section 3 are still valid for higher dimensions. The main open question is whether the expected number of Pareto optimal solutions can still be bounded polynomially for higher dimensions.

Acknowledgements

This work was supported in part by the EU within the 6th Framework Programme under contract 001907 (DELIS) and by DFG grant Vo889/2-1.

References

[1] Pankaj K. Agarwal, David Eppstein, Leonidas J. Guibas, and Monika Rauch Henzinger. Parametric and kinetic minimum spanning trees. In *IEEE Symposium on Foundations of Computer Science*, pages 596–605, 1998.

[2] F. Barahona and W.R. Pulleyblank. Exact arborescences, matchings and cycles. *Discrete Applied Mathematics*, 16:91–99, 1987.

[3] R. Beier and B. Vöcking. Random Knapsack in Expected Polynomial Time. In *Journal of Computer and System Sciences*, volume 69(3), pages 306–329, 2004.

[4] R. Beier and B. Vöcking. Typical Properties of Winners and Losers in Discrete Optimization. In *Proc. of the 36th Annual ACM Symposium on Theory of Computing (STOC-2004)*, pages 343–352, 2004.

[5] Tamal K. Dey. Improved bounds on planar k-sets and k-levels. In *IEEE Symposium on Foundations of Computer Science*, pages 165–161, 1997.

[6] Matthias Ehrgott. *Multicriteria Optimization.* Lecture Notes in Economics and Mathematical Systems Vol. 491. Springer-Verlag, 2000.

[7] P. Hansen. Bicriterion path problems. In *Multiple Criteria Decision Making: Theory and Applications*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, pages 109 – 127, 1980.

[8] Mordechai I. Henig. The shortest path problem with two objective functions. *European Journal of Operational Research*, 25(2):281–291, 1986.

[9] Reiner Horst and Hoang Tuy. *Global Optimization.* Springer-Verlag, 1990.

[10] Naoki Katoh. Bicriteria network optimization problems. *IEICE Transactions Fundamentals of Electronics, Communications and Computer Sciences*, E75-A:321–329, 1992.

[11] K. Mulmuley, U.V. Vazirani, and V.V. Vazirani. Matching is as easy as matrix inversion. *Combinatorica*, 7(1):105–114, 1987.

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

[13] Christos H. Papadimitriou and Mihalis 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 86–92. IEEE Computer Society, 2000.

[14] D. A. Spielman and S.-H. Teng. Smoothed Analysis of Algorithms: Why The Simplex Algorithm Usually Takes Polynomial Time. In *Journal of the ACM*, volume 51(3), pages 385–463, 2004.

[15] George Tsaggouris and Christos Zaroliagis. Non-additive shortest paths. In *Algorithms – ESA 2004*, Lecture Notes of Computer Sciene Vol. 3221, pages 822–834, 2004.

[16] George Tsaggouris and Christos Zaroliagis. Improved FPTAS for multiobjective shortest paths with applications. Technical Report TR-2005/07/03, Computer Technology Institute, Patras, July 2005.

[17] Sergei Vassilvitskii and Mihalis Yannakakis. Efficiently computing succinct trade-off curves. *Theor. Comput. Sci.*, 348(2-3):334–356, 2005.

[18] Arthur Warburton. Approximation of Pareto optima in multiple-objective, shortest-path problems. *Operations Research*, 35(1):70–78, 1987.